ALMA MATER STUDIORUM – Università di Bologna

DOTTORATO DI RICERCA IN
ingegneria elettronica, telecomunicazioni e tecnologie dell'informazione
33° ciclo

Settore Concorsuale: 09/E3 - ELETTRONICA

Settore Scientifico Disciplinare: ING-INF/01 - ELETTRONICA

Ultra-low power IoT applications: from transducers to wireless protocols

Tommaso Polonelli

Coordinatore Dottorato
Prof. Alessandra Costanzo

Supervisore
Prof. Luca Benini

Esame finale anno 2021

"Unless you believe that the mind is separable from the brain both conceptually and empirically – dualism in a strong form – you cannot hope to reproduce the mental by writing and running programs since programs must be independent of brains or any other particular forms of instantiation. If mental operations consist in computational operations on formal symbols, then it follows that they have no interesting connection with the brain; the only connection would be that the brain just happens to be one of the indefinitely many types of machines capable of instantiating the program."

Searle, John. R. (1980) Minds, brains, and programs.

ALMA MATER STUDIORUM – Università di Bologna

Abstract

DOTTORATO DI RICERCA IN

ingegneria elettronica, telecomunicazioni e tecnologie dell'informazione

by Tommaso Polonelli

This dissertation aims to explore Internet of Things (IoT) sensor nodes in various application scenarios with different design requirements. The research provides a comprehensive exploration of all the IoT layers composing an advanced device, from transducers to on-board processing, through low power hardware schemes and wireless protocols for wide area networks. Nowadays, spreading and massive utilization of wireless sensor nodes pushes research and industries to overcome the main limitations of such constrained devices, aiming to make them easily deployable at a lower cost. Significant challenges involve the battery lifetime that directly affects the device operativity and the wireless communication bandwidth. Factors that commonly contrast the system scalability and the energy per bit, as well as the maximum coverage. This thesis aims to serve as a reference and guideline document for future IoT projects, where results are structured following a conventional development pipeline. They usually consider communication standards and sensing as project requirements and low power operation as a necessity. A detailed overview of five leading IoT wireless protocols, together with custom solutions to overcome the throughput limitations and decrease the power consumption, are some of the topic discussed. Low power hardware engineering in multiple applications is also introduced, especially focusing on improving the trade-off between energy, functionality, and on-board processing capabilities. To enhance these features and to provide a bottom-top overview of an IoT sensor node, an innovative and low-cost transducer for structural health monitoring is presented. Lastly, the high-performance computing at the extreme edge of the IoT framework is addressed, with special attention to image processing algorithms running on state of the art RISC-V architecture. As a specific deployment scenario, an OpenCV-based stack, together with a convolutional neural network, is assessed on the octa-core PULP SoC.

Acknowledgements

I am especially indebted to Prof. Luca Benini, Dr. Michele Magno and Prof. Davide Brunelli for their invaluable research support and academic advice during the entire period of my Ph.D. Working in this amazing and stimulating research group has been a reference for professional and personal growth. Moreover, my warmest thanks to Dr. David Boyle that supervised my research activities at Imperial College London.

ad Agata

Contents

List of Figures

List of Tables

Abbreviations

3GPP	**3**rd **G**eneration **P**artnership **P**roject
AAN	**A**rai **A**rui **N**akajama
ABS	**A**crylonitrile **B**utadiene **S**tyrene
ACK	**ACK**nowledgment
ADC	**A**nalog to **D**igital **C**onverter
AI	**A**rtificial **I**ntelligence
ANSI	**A**merican **N**ational **S**tandards **I**nstitute
ASHM	**A**utonomous **S**tructural **H**ealth **M**onitoring
ATWR	**A**symetric double-sided **T**wo-**W**ay **R**anging
BLE	**B**luetooth **L**ow **E**nergy
BN	**B**atch - **N**ormalization
BPM	**B**urst **P**osition **M**odulation
BT	**B**lue**T**ooth
BW	**B**and**W**idth
CL	**CL**uster
CMOS	**C**omplementary**M**etal–**O**xide–**S**emiconductor
CNN	**C**onvolutional **N**eural **N**etwork
cpp	**c**ycles **p**er **p**ixel
CPU	**C**entral **P**rocessing **U**nit
CR	**C**ode **R**ate
CRC	**C**yclic **R**edundancy **C**heck
DL	**D**eep **L**earning
DMA	**D**irect **M**emory **A**ccess
DMIPS	**D**hrystone **MIPS**
DNN	**D**eep **N**eural **N**etwork

DORY	**D**eployment **O**riented memo**RY**
DRM	**D**ynamic **P**ower **M**anagement
DR	**D**ata **R**ate
DSP	**D**igital **S**ignal **P**rocessing
EBR	**E**quivalent **B**it **R**ate
ECG	**E**lectro**C**ardio**G**ram
eDRX	**e**xtended **D**iscontinuous **R**eception
ENOB	**E**ffective **N**umber **O**f **B**its
EPB	**E**nergy **P**er **B**it
EPP	**E**nergy **P**er **P**acket
EPS	**E**nergy **P**er **S**ample
FC	**F**abric **C**ontroller
FDCT	**F**ast **D**iscrete **C**osine **T**ransform
FDM	**F**used **D**eposition **M**odelling
FDMA	**F**requency **D**ivision **M**ultiple **A**ccess
FFD	**F**ully **F**unction **D**evice
FPU	**F**loating **P**oint **U**nit
FSK	**F**requency-**S**hift **K**eying
GBP	**G**ain **B**andwidth **P**roduct
GPIO	**G**eneral **P**urpose **I**nput **O**utput
GPS	**G**lobal **P**ositioning **S**ystem
HD	**H**igh **D**efinition
HPC	**H**igh **P**erformance **C**omputing
I2C	**I**nter-**I**ntegrated **C**ircuit
IC	**I**ntegrated **C**ircuit
IEEE	**I**nstitute of **E**lectrical and **E**lectronics **E**ngineers
IIOT	**I**ndustrial **I**nternet **O**f **T**hings
IPT	**I**nductive **P**ower **T**ransfer
ISA	**I**struction **S**et **A**rchitecture
ISM	**I**ndustrial **S**cientific **M**edical frequency band
JPEG	**J**oint **P**hotographic **E**xperts **G**roup
JTAG	**J**oint **T**est **A**ction **G**roup
LAN	**L**ocal **A**rea **N**etwork

LED	**L**ight **E**mitting **D**iode
LN	**L**i**N**ear
LPL	**L**ow **P**ower **L**istening
LPWAN	**L**ow **P**ower **W**ide **A**rea **N**etwork
LSB	**L**ast **S**ignificant **B**it
LTE	4G **L**ong **T**erm **E**volution
LVDT	**L**inear **V**ariable **D**ifferential **T**ransformer
M2M	**M**achine to **M**achine
MAC	**M**ultiple **A**ccess **C**ontrol
MC	**M**obile **C**omputing
MCU	**M**icro **C**ontroller **U**nit
MEMS	**M**icro **E**lectro-**M**echanical **S**ystem
ML	**M**achine **L**earning
MPU	**M**emory **P**rotection **U**nit
MSE	**M**ean **S**quare **E**rror
NEMO	**NE**ural **M**inimization for pyt**O**rch
NFC	**N**ear **F**ield **C**ommunication
NMI	**N**on-**M**askable **I**nterrupt
NN	**N**eural **N**etwork
OOK	**O**n **O**ff **K**eying
OSI	**O**pen **S**ystem **I**nterconnect
PER	**P**acket **E**rror **R**ate
PAN	**P**ersonal **A**rea **N**etwork
PCB	**P**rinted **C**ircuit **B**oard
PCR	**P**acket **C**ollision **R**ate
PDF	**P**robability **D**ensity **F**unction
PDM	**P**ulse **D**ensity **M**odulation
PEB	**P**ower **E**lectronic **B**oard
PHY	**PHY**sical
PL	**P**acket **L**oss
PRF	**P**ulse **R**epetition **F**requency
PRR	**P**acket **R**eception **R**ation
PSM	**P**ower **S**aving **M**ode

PSNR	**P**eak **S**ignal to **N**oise **R**atio
PULP	**P**arallel **U**ltra **L**ow **P**ower
PW	**P**aging **W**indow
PWM	**P**ulse **W**idth **M**odulation
QNN	**Q**uantized **N**eural **N**etwork
QoS	**Q**uality **o**f **S**ervice
QPSK	**Q**uadrature **P**hase **S**hift **A**rray
QVGA	**Q**uarter **V**ideo **G**raphic **A**rray
RAM	**R**andom **A**ccess **M**emory
RF	**R**adio **F**requency
RFD	**R**educed **F**unction **D**evice
RFID	**R**adio-**F**requency **ID**entification
RGB	**R**ed **G**reen **B**lue
RISC	**R**educed **I**nstruction **S**et **C**omputer
RSSI	**R**eceive **S**ignal **S**trength **I**ndicator
RTC	**R**eal **T**ime **C**lock
RX	**R**eception
SANN	**S**mart **A**gricolture **N**eural **N**etwork
SDADC	**S**igma **D**elta **A**nalog **D**igital **C**onverter
SDK	**S**oftware **D**evelopment **K**it
SF	**S**preading **F**actor
SHM	**S**tructural **H**ealth **M**onitoring
SMA	**S**ub **M**iniature version **A**
SNR	**S**ignal to **N**oise **R**atio
SoC	**S**ystem **o**n **C**hip
SPI	**S**erial **P**eripheral **I**nterface
SPL	**S**ound **P**ressure **L**evel
SWD	**S**erial-**W**ire Debug
TAU	**T**racking **A**rea **U**pdate
TCP	**T**ransmission **C**ontrol **P**rotocol
TI	**T**exas **I**struments
ToF	**T**ime **o**f **F**light
TX	**T**ransmitter

UART	**U**niversal **A**synchronous **R**eceiver-**T**ransmitter
UAV	**U**nmanned **A**erial **V**ehicle
UDP	**U**ser **D**atagram **P**rotocol
UWB	**U**ltra **W**ide **B**and
WBAN	**W**ireless **B**ody **A**rea **N**etwork
WCS	**W**ireless **C**amera **S**ensor
WHT	**W**iFi **H**igh **T**hroughput
WICED	**W**ireless **I**nternet **C**onnectivity for **E**mbedded **D**evices
WLP	**W**iFi **L**ow **P**ower
WPAN	**W**ireless **P**ersonal **A**rea **N**etwork
WPT	**W**ireless **P**ower **T**ransfer
WSN	**W**ireless **S**ensor **N**etwork
WUA	**W**ake **U**p **A**udio
WUR	**W**ake **U**p **R**adio

Chapter 1

Introduction

The evolution of the Internet-of-Things (IoT) brings new classes of applications and improvements in the efficiency of existing industrial and daily life support services. Application-specific requirements, as well as sensing and connectivity ability of devices, have introduced new challenges for IoT developers. With the progressing expansion of information technology, the IoT has evolved to play an essential role in our daily lives, often more than we perceive. Indeed, wearable sensors for healthcare, smart cities, smart transportation, and logistics are already manufactured on a large-scale [1]. Interconnected sensors/things can collect and exchange different information amongst themselves through advanced network infrastructure connected by millions of IoT nodes [1]. A diversity of IoT employments can provide more accurate and more fine-grained services for final users or other machines working at a logic upper layer. For this purpose, more and more sensors and smart devices are being deployed, which will generate massive data, terabytes per seconds, demanding further computational processing and enhancing knowledge for both service providers and users [2]. This work aims to provide a detailed overview of IoT technologies required from an embedded design perspective and specific properties associated with IoT sensors in embedded systems' landscape. It investigates essential technologies for the development of IoT systems, edge computing, low-power transducers, and in-field performance biased by low-power constraints.

Recent and ongoing advances in wireless communication, ultra-low power processors, sensors and actuators, smart power management, energy harvesting, and cloud/fog computing have enabled the emergence of IoT [3]. While new Wireless Sensor Network (WSN) protocols enable low-cost communication and wide-area coverage, ultra-low-power system-on-chips (SoC) support portable embedded devices. On the other side, cloud and fog computing offer HPC resources and services to the local or global servers, providing additional resources and user-friendly graphical interfaces for large-scale handling of data [4]. IoT is generally characterized by wireless connectivity, the principal difference from conventional embedded devices. In a broader sense and vision, the Internet of Things is a global infrastructure of heterogeneous, networked embedded devices and objects [1]. Edge devices' communication ability lets sensors and smart things interact and cooperate with other machines and users [2–5]. IoT infrastructure covers several application domains that include wireless sensor networks [6], machine-to-machine (M2M), Mobile Computing (MC), real-time health monitoring [7], as well as wearables [8], agriculture [9], maintenance [10], and structural health monitoring (SHM). There have been many reseaincreasedrch efforts on IoT from the perspective of networking, power consumption [6], data access (security and privacy) [11]; however, it has gained growing attention from the perspective of low-power computing, also known as edge computing [1].

In conventional IoT frameworks, all data must be uploaded to centralized servers, and after computation, a result may be sent back to the sources. This process could create significant pressure on crowded networks, specifically in data transmission costs in bandwidth and energy. Moreover, the performance of the system will worsen with increasing data size. For IoT applications that are time-sensitive, this effect is evidenced when very short response times are non-negotiable, for example, in smart transportation [12], autonomous driving [13] in which conventional cloud computing-based service cannot satisfy QoS requirements. Since the computation process is executed on the cloud, and massive data transmissions occupy the limited bandwidth and network resources, then the result will be significant latency in the network, which is unacceptable for time-sensitive services, such as mobility automation where a millisecond response must be guaranteed. This is a severe problem for IoT, as these applications will have an impact on safety and emergency response and it will heavily affect the system scalability in non-time-sensitive applications. Furthermore, most IoT devices have limited power. To extend their lifetime, it is necessary to balance power consumption by scheduling on-board computation and remote transmission, which is typically the most power-hungry part of an IoT device [14]. Besides, processing data in computation nodes with the shortest distance to the user will reduce transmission time, and so, the response delay [15]. In a cloud computing-based service, the data transmission speed will be affected by network traffic, increasing power consumption costs. Thus, scheduling and processing allocation is a critical issue that must be considered [15].

Nowadays, researchers and industries are pushing in the direction to exceed traditional computation based on stationary workstations [16]. Indeed, the IoT is rapidly merging into daily life, envisioning that most electronic devices, such as smartphones, vehicles, and any other embedded devices, will be connected with remote data centers, exchanging ever more information [1]. Economic studies report the IoT market as one of the most burgeoning sectors that will impact US interests in 2025 [17]. Likewise, the number of interconnected devices exceeded the world's human population in 2012, when the number of IoT devices reached 9 billion [16].

The IoT structure is mainly composed of three different communication models: M2M, machine to cloud and machine to gateway. The former allows devices to directly connect and exchange information among each other, without any intermediate assistance. This topology type often requires complex protocols and interfaces, making it challenging to work with low-power constraints. In [18], the authors show that one of the most used mesh protocol for IoT (ZigBee [19]) needs at least 5× more energy than LoRaWAN [20]. In this field, the new IETF IPv6 over the TSCH mode (6TiSCH) has standardized a set of protocols to enable low power mesh industrial-grade networks [21]; however, today there are not practical comparisons among ZigBee and other mesh systems, so it is not considered in this dissertation. Machine to cloud is one of the most used methods, but it requires the support of computer network protocols, such as WiFi, Ethernet, and TCP/IP layers, which are not suitable for ultra-low-power devices with a targeted lifespan of years. The latter, machine to gateway, suits the most stringent energy restrictions; indeed, it assumes the presence of a gateway, which acts as an intermediate bridge between a lightweight communication protocol and the cloud throught an internet connection. It is clear that, for low-power IoT applications, the star network topology provides the lowest cost per bit in wireless remote communication. Besides others, such as mesh and tree typology that provide dedicated low energy stacks, the star architecture can be specifically optimised for point-to-point connection by relying on a gateway with high capabilities, which can support communication lacks from the IoT device. For example, LoRaWAN [20] employs sophisticated gateways to support extremely simple sensor node, which are not even able to avoid channel conflicts with other devices.

Typically, the IoT network features three types of components [1]: sensors/devices, gateways/local network, and a remote server/cloud, representing the data source/actuators, communication networks, and data processing, respectively. Similarly, edge computing structure can be split into three sub-blocks [1]: front-end, near-end, and far-end. Sensor nodes are the key component of the IoT framework, and they produce the majority of measurement data in the networks, and, in some cases, they also manage actuators. These devices usually provide diverse types of data from various kinds of transducers, from temperature to CMOS cameras. In addition, they are often battery supplied and deployed in remote and extreme environments. These end devices are deployed at the front-end of the edge computing structure, hence providing more interaction and better responsiveness for the end-users applying the edge computing paradigm [22], which is a distributed computing structure that brings computation and data storage closer to the location where it is needed or collected, to improve response times and save bandwidth. Indeed, via the on-board computing capacity provided by the plethora of SoCs, edge computing can provide real-time services for some applications or, more frequently, decrease the upload operations to the cloud [1]. Nevertheless, most requirements cannot be performed at the front-end environment due to the limited computational capacity. Thus, in these cases, the end devices need to forward the resource to the server.

Since the gateway can also have numerous features, such as hundreds-MHz CPU, data caching, and a reliable power supply, in edge computing most of the data computation and storage will be migrated to this near-end machine [23]. In doing so, the whole application can achieve much better performance on data computing and scalability, with a small increase in the gateway cost. Via backhaul networks, cloud servers will receive the data directly from sensors or through gateways [22, 23]. For most IoT applications, the cloud servers are programmed to accomplish the whole data processing tasks; then, they must have a significant capacity for computation and storage to satisfy different requirements. As the cloud servers are the third layer farther away from the front-end, the transmission latency cannot be considered as negligible. Nonetheless, the cloud servers in the far-end environment can provide more computing power and more data storage. For example, they can provide massive parallel data processing, data mining, data management, and machine learning [22, 23].

The development of a robust, low-power, low-cost, and highly scalable WSN is still an open challenge. Designing and implementing highly energy-efficient IoT sensors requires a multi-level approach to balance the limitations imposed by applications and technology specifications.

1.1 Thesis outline and contribution

Aiming at an even more connected world, the following sections expose the IoT framework analyzing and designing the most recent trends and technologies, providing an overview of main open challenges and limitations. This thesis, structured as a development and design guideline, illustrates the five primary blocks that compose a standard IoT node: transducers, power management, edge computing, and low power wide area network protocols and schemes.

The diversity of IoT applications and technologies makes it difficult to present a comprehensive general statement for IoT requirements in hardware and software. Therefore,

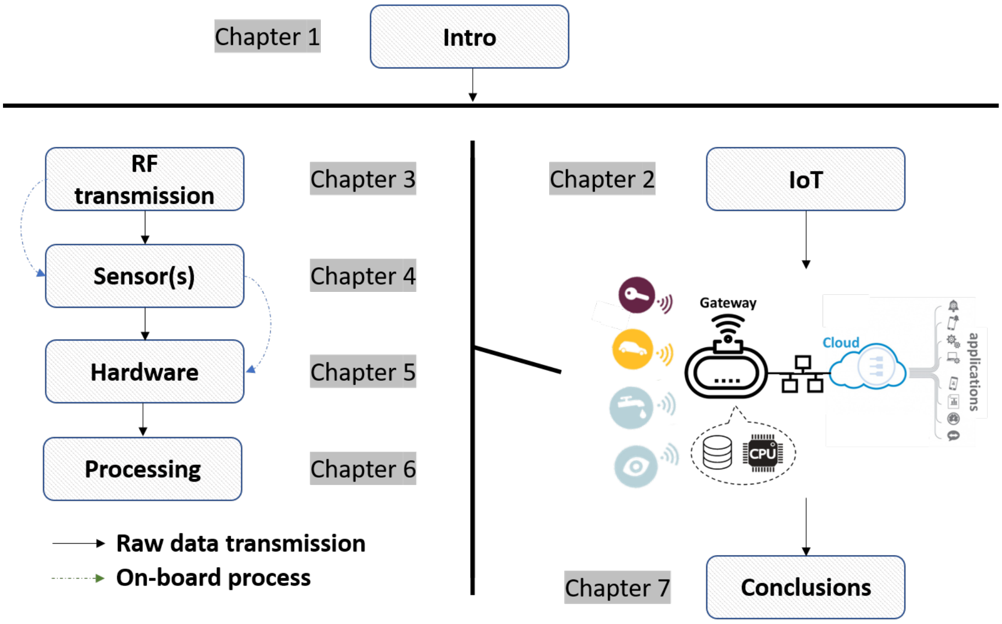


Figure 1.1: Thesis outline

the IoT embedded designer faces questions whose answers are challenging as the solutions can be contradictory, e.g., which wireless communication technology *i*) covers the required range, *ii*) provides the required data rate, *iii*) is still (ultra) low-power and meets the energy constraints? What trade-offs to make between *i*) hardware functionalities and energy consumption, *ii*) on-board processing and computation offloading, etc.? How to handle the uncertainty and unpredictability of IoT systems?

Fig. 1.1 presents the thesis structure, focusing on the technical part. It is sorted following the planning and design of a new IoT application, in which the wireless connectivity is often a project parameter; hence affecting sensors and components selection. Depending on project objectives, the processing part might be left out if a raw data transmission is preferable. Other than Chapter 1, which introduces this thesis, the IoT framework is described in Chapter 2. An analysis of the most common wireless protocols is presented in Chapter 3. Chapter 5 shows a design of a custom transducer and a comprehensive IoT sensor node project. Chapter 6 proposes an example of low-power hardware design for general purpose sensors, providing an energy model to estimate the average battery lifetime depending on specific uses cases. As already mentioned, the edge computing paradigm is evolving and is increasingly necessary; hence Chapter 7 shows two use cases in which a complex application, such as image processing and successive feature extraction, can run on a constrained and low-power MCU. Finally, Chapter 8 concludes the thesis.

1.2 List of publications

The content of this thesis and the main contributions have been published to a large extent in international conferences and journals. The corresponding chapters are indicated in parentheses.

1. Polonelli, T., Brunelli, D., Benini, L. (2018, October). Slotted aloha overlay on lorawan-a distributed synchronization approach. In 2018 IEEE 16th international conference on embedded and ubiquitous computing (EUC) (pp. 129-132). IEEE. (*Chapter 3.1*) [24]
2. Polonelli, T., Brunelli, D., Marzocchi, A., Benini, L. (2019). Slotted aloha on lorawan-design, analysis, and deployment. Sensors, 19(4), 838. (*Chapter 3.2*) [25]
3. Polonelli, T., Brunelli, D., Guermandi, M., Benini, L. (2018, September). An accurate low-cost Crackmeter with LoRaWAN communication and energy harvesting capability. In 2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA) (Vol. 1, pp. 671-676). IEEE. (*Chapter 5.1*) [26]
4. Polonelli, T., Brunelli, D., Girolami, A., Demmi, G. N., Benini, L. (2019, June). A multi-protocol system for configurable data streaming on IoT healthcare devices. In 2019 IEEE 8th international workshop on advances in sensors and interfaces (IWASI) (pp. 112-117). IEEE. (*Chapter 3.10*) [27]
5. Ballerini, M., Polonelli, T., Brunelli, D., Magno, M., Benini, L. (2020). NB-IoT vs. LoRaWAN: An Experimental Evaluation for Industrial Applications. IEEE Transactions on Industrial Informatics. (*Chapter 3.7*) [28]
6. Ballerini, M., Polonelli, T., Brunelli, D., Magno, M., Benini, L. (2019, July). Experimental evaluation on NB-IoT and LoRaWAN for industrial and IoT applications. In 2019 IEEE 17th International Conference on Industrial Informatics (INDIN) (Vol. 1, pp. 1729-1732). IEEE. (*Chapter 3.9*) [29]
7. Polonelli, T., Brunelli, D., Bartolini, A., Benini, L. (2018, September). A lorawan wireless sensor network for data center temperature monitoring. In International Conference on Applications in Electronics Pervading Industry, Environment and Society (pp. 169-177). Springer, Cham. (*Chapter 3.1*) [30]
8. Polonelli, T., Battistini, D., Rusci, M., Brunelli, D., Benini, L. (2019, September). An Energy Optimized JPEG Encoder for Parallel Ultra-Low-Power Processing-Platforms. In International Conference on Applications in Electronics Pervading Industry, Environment and Society (pp. 125-133). Springer, Cham. (*Chapter 7.1*) [31]

9. Polonelli, T., Qin, Y., Yeatman, E. M., Benini, L., Boyle, D. (2020). A Flexible, Low-Power Platform for UAV-Based Data Collection From Remote Sensors. IEEE Access, 8, 164775-164785. (*Chapter 6*) [32]
10. Brunelli, D., Polonelli, T., Benini, L. (2020, October). Ultra-low energy pest detection for smart agriculture. In 2020 IEEE Sensors (pp. 1-4). IEEE. (*Chapter 7*) [33]

Chapter 2

Sensor nodes for IoT

An IoT sensor comprises many functional blocks to enable various services, such as sensing (On-board - Fig. 2.1), identification, actuation (Off-board - Fig. 2.1), communication, and management (Power Unit and memory - Fig. 2.1). It can exchange data with other connected devices and remote servers, or perform some tasks locally. An IoT device usually supports several interfaces for communications to other things, both wired and wireless. These include interfaces for internal sensors, interfaces for Internet connectivity or LPWAN, high-speed memory and storage, and audio/video drivers. IoT devices can also be of varied shapes and dimensions, such as smartwatches, wearable sensors, automobiles, and industrial machines. The communication block (RF - Fig. 2.1) manages the connection among devices and remote servers. IoT communication protocols generally work using a physical link layer, MAC (Media Access Control) layer, and application layer. Moreover, this block has to be flexible to support various functions such as services for data transfer, device control, data publishing, data analytics, and device discovery. The management block (Processing - Fig. 2.1) usually takes care of the protocol stack, sensor acquisition, and low-power states. At the same time, it provides functions such as authentication, authorization, privacy, message integrity, content integrity, and data security. Finally, the application layer is the most important, as it acts as an interface that controls and monitors various aspects of the IoT framework.

2.1 Low-power wireless devices for IoT

A wireless sensor network consists of mostly tiny, low-cost, resource-constrained, and low power sensor nodes organized to work cooperatively [34]. Fig. 2.1 shows the main parts of a sensor node.

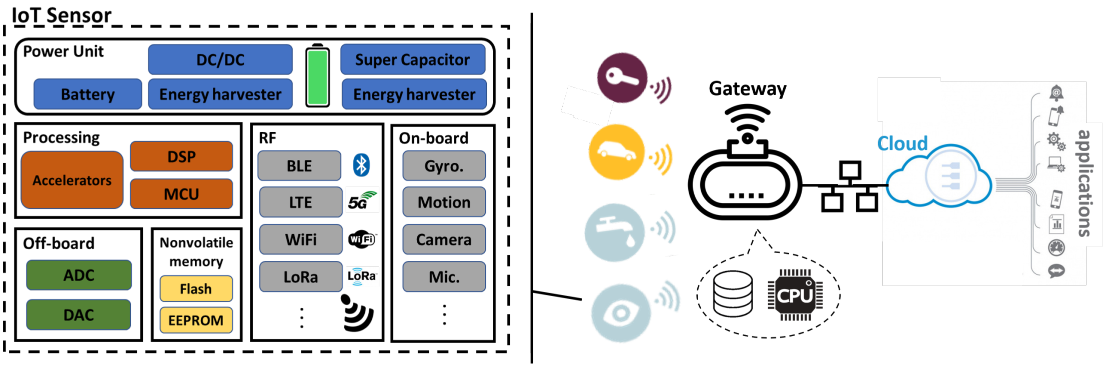


Figure 2.1: Top-level overview of a generic IoT sensor node

Several multi-functional devices continuously work without any interruption in a specified area for specified tasks. As part of the fundamental system feature, the power manager controls and manages the operation states and power down states of the main sub-blocks, e.g., the MCU, volatile and non-volatile memories, and the RF transceiver. Its primary goal is to reduce the average energy consumption in both active and sleep modes. It is mainly composed of energy sources, such as batteries, supercapacitors, energy harvesters, and low-power low-leakage non-linear voltage converters, which supply stable and low-noise voltage for analog and digital parts. Moreover, due to tiny and constrained power sources, the power unit must support current spikes often generated during RF transmissions. Dynamic Power Management (DPM) is referred to software, hardware algorithms/techniques which control the energy and performance trade-off of a low power system, hence achieving longer battery lifetime [35]. The power dissipation of the sensor node can be modeled using static and dynamic methods; both of them are useful to describe a particular system state. Leakage current flow in ultra-low power components can be modeled at design time by using static techniques, whereas the dynamic power is the result of digital switching components or RF spikes that must be modeled using dynamic methods and in-field measurements.

In Fig. 2.1, the processing block includes an SoC and its external components. This block is usually defined as the core of the systems and manages all the sensors and the radio interface. Depending on the board's complexity, it could require external components such as RAM, crystals, and an RTC. Almost every sensor node on the market includes an on-boards sensor directly on the main PCB; this method drastically reduces the overall production cost and decreases the object volume. The most widespread ICs are accelerometers, gyroscopes, microphones, and CMOS cameras.

The RF subsystem is responsible for the wireless communications of the sensor node. The transceiver has mainly four operational states which are Receive, Transmit, Idle, and Sleep. Radio Frequency (RF), Backscattering, Infrared, and acoustic waves can

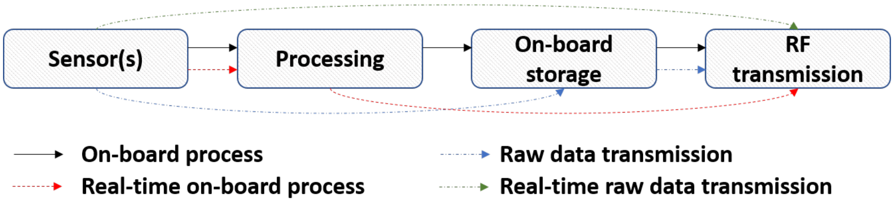


Figure 2.2: General stages of a sensor node for IoT

be chosen in transceiver technology as a wireless media. Among these options, RF is widely preferred for WSNs. In this topic, a plethora of standards are competing in the industrial and healthcare market, a subject further discussed in section 2.3.

IoT general operation stages include data acquisition, data processing, data storage, and data transmission. The first and last stages exist on every application, while the processing and storage are dependent on the project specifications. Fig. 2.2 summarises the workflow of a typical device. Low power ICs with MEMS technology can generally reduce the energy consumption of data acquisition. On-chip gyroscopes, accelerometers, microphones, and biosensors are the most commonly used in general applications. However, to collect minimum data as possible, an energy-efficient sensing method exploits the input spatial data and temporal characteristics while maintaining the required signal quality [2]. Reducing the amount of input data is fundamental to decrease the energy for processing, storage, and transmission. One of the primary purposes of on-board processing is to reduce the amount of data transmitted remotely, extracting features directly on the network edge, or packing the raw data using lossless and lossy compression algorithms. For example, compressed sensing technique reconstructs the original signal from much far less samples than Nyquist theory exploiting the sparseness property of IoT data at the cost of accuracy loss[36].

2.1.1 Power Unit and Energy Harvesting

When running on batteries, the sensor nodes' limited lifetime is one of the main challenges in implementing WSNs [37]. In many IoT applications, sensor nodes are deployed in places where access may be impossible for human operators, making it difficult to change the batteries of the nodes regularly. Hence once a sensor node drains its battery, that node can be considered out of service. Over the years, researchers have focused on designing energy-saving techniques to minimize sensor nodes' energy consumption at the physical, communication, and routing layers [38]. However, even with these precautions, once the battery energy is exhausted, the sensor node can no longer participate in the network operations [37]. To address the finite supply time challenge, new WSN platforms support the harvesting of energy from the immediate surroundings.

These devices can capture small amounts of energy from heat, vibration, light, sound, or movement within the background [39]. By recharging the battery and developing energy harvesting aware algorithms that support the energy-neutral operation, WSNs could theoretically have an infinite lifetime. Several researchers have explored technologies to harvest energy from common natural sources, such as the sun, water flow, and wind. In contrast, others have explored human-made energy sources, such as human walking, magnetic fields, high-frequency vibrations, and RF fields [40].

Recent advances in this technology, and the commercialization of devices that support energy harvesting, have driven to a new class of WSNs, referred to as Energy-Harvesting WSNs (EH-WSNs) [41]. The ability of EH-WSNs to power sensor nodes through ambient energy sources has led to a shift in the design requirements and goals of these networks, as network lifetime is, ideally, no longer an issue. Instead, the goal in EH-WSNs is to support energy-neutral operation, so that the EH-WSN can operate continuously using the harvested energy. In particular, rather than reducing node energy consumption to extend network lifetime, in EH-WSNs, it is essential to re-consider the impact of performance metrics such as energy-efficiency, fairness, scalability, and latency in the presence of energy flow into the network. Similarly, physical, MAC, and routing protocols need to be re-designed to optimize the frequency at which the energy is used, rather than simply minimizing the total energy expenditure. There are multiple sources of natural and human-made energy that can be harvested by the sensor node, where each of them provides a different power density, as shown in Table 2.1.

Table 2.1: Energy-Harvesting Sources and their Corresponding Power Densities

Energy Source	Types	Energy Harvesting Method	Power Density
Radiant	Solar	Solar cells (indoor)	$10\mu W/cm^2$
		Solar cells (outdoors)	$15mW/cm^2$
	RF	Electromagnetic	$0.1\mu W/cm^2$
		Electromagnetic	$0.01\mu W/cm^2$
Mechanical	Wind flow	Electromagnetic	$16\mu W/cm^3$
	Acoustic noise	Piezoelectric	$960nW/cm^3$
	Motion	Piezoelectric	$330\mu W/cm^3$
Thermal	Body heat	Thermoelectric	$40\mu W/cm^3$

[37, 41, 42]

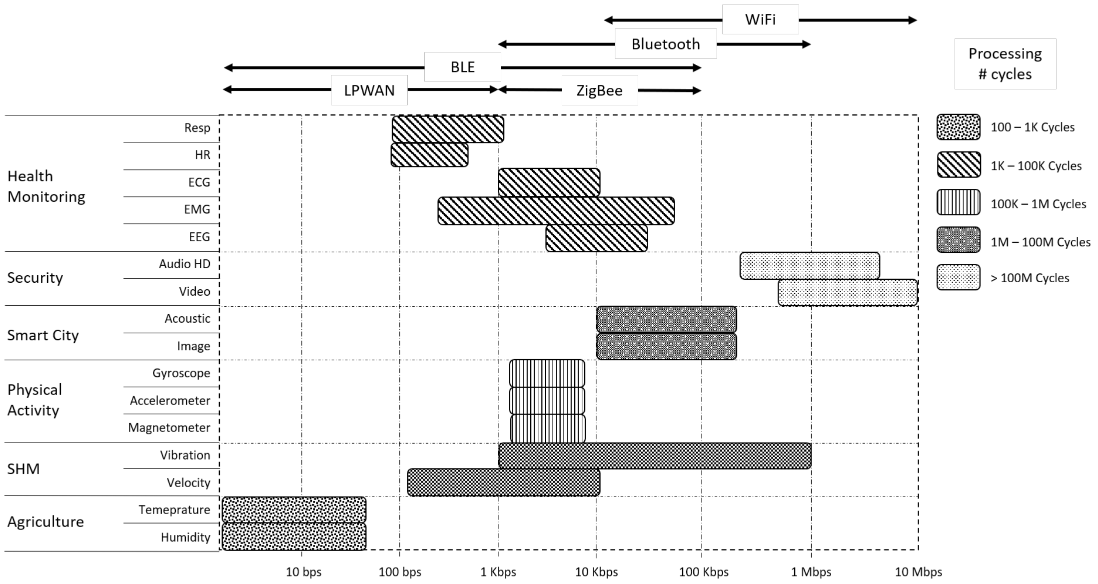
2.1.2 Sensors and Transducers

One of the most crucial parts of an IoT device is the sensor block [43]. Indeed, transducers convert environmental phenomena like light, humidity, heat, and sound into electrical signals. In recent years, the rapid development in multiple sensing technologies has eased how electronics perceive external conditions. In the following, a list of the most common sensors on the market is presented. Proposing an exhaustive overview of all the transducers and sensors present on the market goes beyond this work's scope. Every day, a new technology or IC is presented, and others become obsolete.

- Micro-Electro-Mechanical Systems (MEMS) such as accelerometers, gyroscopes, acoustic sensors, magnetometers, chemical sensors, pressure sensors, and piezoelectric sensors. They are widely used in medical and fitness devices, industrial machinery, wearable devices, navigation, fall detection, or SHM of machinery or buildings [44]. Moreover, MEMS microphones are widely used for security and surveillance, as well as for smart cities [44].
- Piezoelectric transducers such as shock, vibration, and pressure.
- CMOS-based sensors include cameras, humidity, temperature, chemical composition sensors, and all the capacitive proximity sensors. Besides standard imaging sensors, IR are emerging in IoT demand to measure the objects' temperature differences, useful for security applications to detect intruders, or get thermal images.
- Light, LED-based sensors such as proximity and ambient light transducers.
- Location sensors base their technology on GPS and ToF measurements, such as ultrasonic transducers and the UWB ranging protocol. Industry and smart transportation use this sensor for tracking and localizing the objects.
- Biosignals contain qualitative and quantitative information about the human body's state, which has been exploited in the past to improve the diagnosis and treatment of health disorders. The body can provide a variety of signal sources that include, for instance, electrocardiogram (ECG), electrooculogram (EOG), and galvanic skin response (GSR).
- Chemical Sensors are mainly used to measure air quality and biosignals (e.g., CO, NO2, SH2, and CO2) in healthcare applications, environment monitoring, or smart factories [45].

These advancements, deeply discussed in [45–51], are the key to the widespread use of IoT platforms in daily life. Each transducer or sensor listed above generates a variable quantity of data, spanning between few bytes per day (temperature) to GB per day (HD CMOS camera). Then, Table 2.2 plots the typical data generation rate of different sensors according to the reported application scenarios. The required wireless bandwidth is annotated, as well as the estimation of the needed processing time to perform the typical actions on the captured data. As noted, the selection of the sensor(s) is fundamental to keep computational requirements and energy consumption under control. Indeed, most of the hardware requirements and limitations come from the sensor itself. A high-resolution CMOS camera cannot be managed by a tiny 8-bit MCU and 1 KB of RAM. Hence the sensor choice is one of the most crucial aspects designing new IoT solutions. In [52, 53], a low resolution (QVGA) grey-scale camera compensates the constrained resources of the MCU and the battery. Moreover, the device's price is strongly influenced by the transducer, contributing up to 60% [52] of the final cost. For this reason, the growing demand for low-cost and low-power sensors has resulted in a broad use of IoT technology. As a contribution of technological research, Section 5.1 presents and evaluates an innovative transducer optimized to enhance the sensitivity/cost ratio.

Table 2.2: Data range generation and number of processing cycles by different sensors in typical IoT application



2.1.2.1 Himax HM01B0

Results presented in this work make use of a specific and ultra-low power QVGA CMOS image sensor worth introducing. The HM01B0[1] from Himax Imaging [54] is an ultra-low-power sensor that enables the integration of an *always on* camera for computer vision and edge computing such as gestures, intelligent ambient light, proximity sensing, tracking, and object identification. It consumes a very low power of $< 2\ mW$ at QVGA 30 fps. These features make this camera the perfect pair for IoT low-power sensor nodes. The HM01B0 has a 320 × 320 pixel resolution and supports a 320 × 240 window mode,

---

[1] https://www.himax.com.tw/products/cmos-image-sensor/image-sensors/hm01b0/

which can be read out at a maximum frame rate of 60 fps, and a 2 × 2 monochrome binning mode with a maximum frame rate of 120 fps [54]. The video data is transferred over a configurable 1 bit, 4 bit, or 8 bit interface. The sensor integrates a black level calibration circuit, automatic exposure and gain control loop, self-oscillator, and motion detection circuit.

2.1.2.2 LVDT

The Linear Variable Differential Transformer (LVDT) transducer accurately measures the movement between the spring-loaded sliding armature and the exterior body. These rugged and self-contained units are ideal for recording structural members' displacements due to dynamic loads and temperature variations.

Although technologies like global positioning system (GPS) help determine global changes in position, LVDTs and linear potentiometers help in traditional displacement measurements. Usually, these are deployed between two locations or at the boundary of the structure for measuring relative displacements. An LVDT sensor can determine the movement in one direction between two points. LVDT is quite common to measure displacement. Often, LVDTs are used to verify new displacement monitoring systems' accuracy and prove to be very accurate compared to these other methods [26]. To determine the structure long-term degradation, crack opening displacements can be measured directly using LVDT because of its long-term stability [26].

This sensor is used in Section 5.5.3 as a reference for a custom made transducer.

2.2 Processing Unit

Miniaturization of electronic devices allows for a paradigm shift from external data analysis and classification to on-board processing. Hence, the computational unit plays a growing role in modern IoT and edge computing. Nevertheless, porting applications, for both cloud and server, to low-cost embedded systems are not trivial tasks as the final system requires both computational power (real-time constraints) and energy efficiency (battery life). With this goal, this work focuses on commercially available platforms such as the ARM Cortex-Mx, which provides a compatible power budget while delivering the required computational capability. A step forward has been achieved later through GAP8, a Parallel Ultra-Low Power (PULP) architecture from Greenwaves. This section introduces the general characteristics of both architectures.

2.2.1 ARM Cortex-M4 SoC

The ARM Cortex-M4 is a 32-bit RISC microcontroller that includes DSP instructions and a floating-point unit, making it suitable for edge computing at low energy consumption. Several IoT systems have adopted this architecture [27, 55–57], to allow for online processing.

Throughout this work, STMicroelectronics STM32F4 and STM32L4 series have been used extensively. The processor is based on Harvard architecture with three-pipeline stage and branch speculation, implementing 32-bit RISC ISA. It also includes the entire 16-bit $Thumb^{®} - 1$ and 16/32-bit $Thumb^{®} - 2$ instruction sets. It includes an FPU for fast floating-point operations and it is equipped with more than 128 kB of SRAM and 1 MB of non-volatile flash memory. The STM32F4 series features a power density of 238 μW/MHz, offering up to 210 DMIPS and 566 CoreMark at 168 MHz on general-purpose applications; whereas, STM32L4 is more suitable for low-power applications, with a power density of 129 μW/MHz but lowering the computational performances, with 100 DMIPS at 80 MHz.

Their rich set of peripheral (SPI, I2C, UART, PDM, ADCs) allows flexibility to communicate with external devices and transducers such as specialized sensors for each targeted application. It features non-maskable interrupts (NMI) and up to 240 physical interrupts, with different priority levels (from 8 to 256). It also offers several power modes, allowing to reduce the overall power consumption of the system.

Moreover, it features a JTAG and Serial-Wire Debug (SWD) ports with up to eight breakpoints and four watchpoints. The processor also includes Memory Protection Unit (MPU) for the eight memory sub-regions, preventing access to privileged application data.

2.2.2 PULP platform

PULP platform is an open-source project created by the University of Bologna and ETH Zurich. PULP is a many-core platform (OpenRISC or RISC-V ISA) able to operate over a broad range of voltages and frequencies, which provide energy efficiency for both low and computationally heavy applications.

Several versions of the PULP architecture have been developed, and they differ on the chip technology (CMOS, FD-SOI implemented on 65, 40, or 28 nm), the number of processing units, and the memory size. In the following, GAP8 is introduced as it has been included in the systems and applications presented in this work.

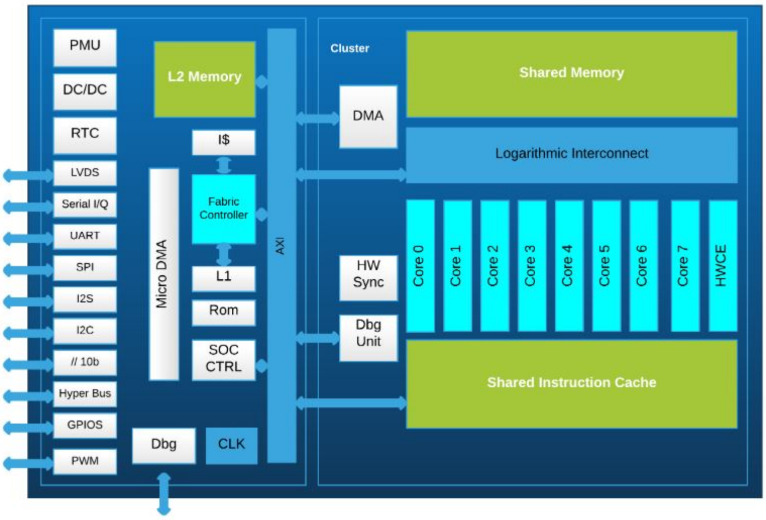


Figure 2.3: GAP8 Layout from Greenwaves Technologies[2]

2.2.2.1 GAP-8

GAP-8 is a multi-core processor derived from the PULP open-source computing platform [58], fabricated in TSMC 55 nm LP CMOS technology. It leverages the RISC-V ISA's flexibility and openness to integrate a state-of-the-art microcontroller, several peripherals, associated with a powerful programmable parallel processing engine for flexible multi-sensor (image, audio, inertial) data analysis and fusion, which includes a dedicated convolutional engine for deep neural network inference. GAP-8 can deliver up to 10 GMAC/s for CNN inference (90 MHz, 1.0 V) within a worst-case power envelope of 75 mW [58]. Fig. 2.3 provides a schematic view of the GAP-8 SoC architecture, including voltage and frequency domains, the Fabric Controller (FC), and the Cluster (CL). Table 2.3 shows the main features and CMOS technology. The Fabric Controller is an MCU based on an RISC-V architecture extended for energy-efficient digital signal processing enhanced with an instruction cache and a fast access data memory. The FC manages a full set of peripherals (i.e., QSPI, I2C, I2S, CAM), enabling parallel capture of images, sounds and vibrations, and a 4-channel PWM interface for motor control, for applications such as machine learning and data processing on edge. High-speed data transfers between the peripherals are managed by a low-power DMA to minimize the number of interactions and the workload. The L2 memory (512 kB) is available on the SoC, which stores the primary boot-code and peripheral data. GAP-8's memory hierarchy is organized as a multi-layer structure. Every single core in the chip can access all memory locations with progressively increased access latency for L2 memory or external DDR memory (L3). To overcome the latency of L3 and L2 memory, the cluster contains a dedicated multi-channel DMA capable of 1D and 2D mass memory

---

[2] https://greenwaves-technologies.com/manuals/BUILD/HOME/html/index.html

transfers. The cluster is used when applications running on the fabric controller offload highly computation-intensive kernels. It contains 8 RISC-V cores identical to those used in the FC, allowing the SoC to run the same binary code on either the fabric controller or the cluster. The CL is served by a shared L1 data memory supported by a highly optimized interconnect between the cores' load/store units and the memory banks. The shared L1 can serve all requests in parallel with a single-cycle access latency and a $< 10\%$ average contention rate. Fast event management and synchronization are supported by a dedicated hardware block, enabling fine-grained parallelism and high energy efficiency. This block also controls the top-level clock gating of every single core in the cluster. Hence a core waiting for an event is instantly brought into a fully clock gated state, zeroing its dynamic power consumption.

Table 2.3: Mr.Wolf SoC features.

Technology	CMOS 55 nm LP
Chip Area	10 mm^2
Memory Transistors	576 kB
Equivalent Gates (NAND2)	2 Mgates
Voltage Range	0.8 V – 1.2 V
Frequency Range	32 kHz – 250 MHz
Sleep Power (State Retentive)	30 μW
SoC Power Density @ 1.2V	156 μW/MHz
Cluster Power Density @ 0.8V	300 μW/MHz

2.3 Wireless communication standards

Communication protocols establish the backbone of IoT systems and enable connectivity and coupling to distributed applications. Their primary purpose is to allow devices to exchange data over the network defining exchange formats, encoding, addressing schemes, and network routing from source to destination [5]. Other functions include sequence control, flow control, and retransmission of lost packets. Fig. 2.4 compares different wireless communication technologies concerning coverage and data rate, while Table 2.4 shows some characteristics which are needed to be considered in the design process of IoT devices.

Table 2.4: Wireless communication standards for IoT

		Bluetooth	BLE	ZigBee	WiFi	LPWAN	UWB	Cellular	
								LTE	NB-IoT
Range	indoor	1 - 10 m	1 - 10 m	20 m	20 m	10 km	20 m	5 km	10 km
	outdoor	1 - 100 m	1 - 20 m	1500 m	100 m		~ 100 m		
Bit rate [Mbps]		1 - 3	1	0.25	100	0.05	6	75 - 300	0.2
Throughput [Mbps]		1.5	0.3	0.15	50	0.05	5.8	NA	
Carrier Freq. [GHz]		2.4	2.4	2.4	2.5/5	sub-GHz	4/6	2.1	0.8-1.9/2
Network topology		star	star/mesh	mesh	star	star/p2p	star/p2p	NA	

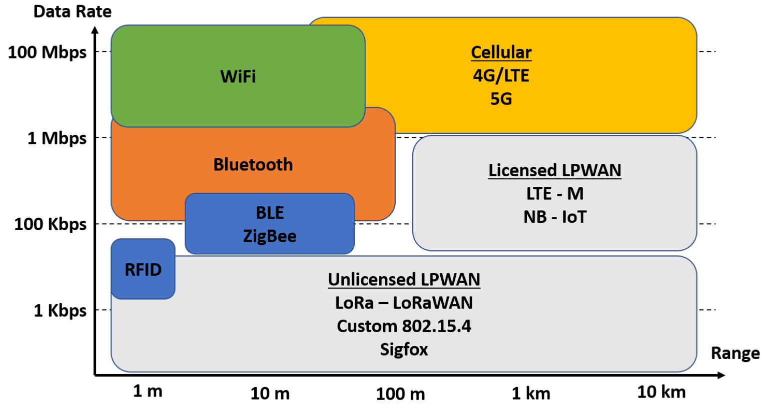


Figure 2.4: Wireless communication standards for IoT

Table 2.5: Common protocols grouped by IoT application domains

		Application domains					LPWAN
		Healthcare	Smart Cities	Smart Building	Automotive	Industry	M2M
Protocols	Bluetooth	very high	low	high	medium	low	high
	BLE	very high	low	medium	very low	low	high
	ZigBee	medium	high	very high	very low	high	medium
	WiFi	low	high	medium	medium	low	high
	LPWAN	low	very high	high	high	high	very high
	UWB	medium	very high	medium	very low	high	low
	LTE	low	very high	low	high	medium	low
	NB-IoT	very high	very high	medium	high	high	low

Most of the connectivity requirements depend on design and customer constraints. For example, by increasing/decreasing the RF power, the transmission range can be further increased/decreased, also affecting the current consumption. Each of these protocols has its advantages and disadvantages [1]. For instance, WiFi, BLE, and ZigBee may suffer interference due to the coexistence of the same frequency band (i.e., 2.4 GHz) in a crowded environment. A high packet loss probability can lead to a severe drop in data rate, which consequently may increase the energy consumption of low-power devices. Hybrid transmission schemes could be the best-fitted solutions for IoT applications. The evolution of multi-protocol integrated transceivers has opened a room for more efficient wireless communication. For example, BLE does not fit continuous data streaming applications, but it is highly efficient for sending small, discrete data chunks. On the other hand, Bluetooth 2.0 and WiFi offer higher throughput and efficiency for data streaming with higher data rate demands. Hence, with a multi-protocol transceiver, the control units can enable the best technology for the application requirements, keeping unused peripherals in deep sleep mode. To provide a better overview of the trade-off between different wireless communication technologies, Table 2.5 shows the suitability of each application domain.

---

[2] https://www.ti.com/wireless-connectivity/multi-protocol/overview.html

2.3.1 WiFi

The WiFi (IEEE 802.11) is a collection of Wireless Local Area Network (WLAN) communication standards. For example, 802.11b and 802.11g operate in the 2.4 GHz band, 802.11a and 802.11ac operate in the 5 GHz band, 802.11n operates in the 2.4/5 GHz bands, and the newest 802.11ad explores the 60 GHz band. These standards provide wide baud rate settings, from 1 Mbps to 6.75 Gbps. In Europe, the IEEE 802.11 supports communication range between 20 m (indoor) to 100 m (outdoor).

2.3.2 WPAN

IEEE 802.15.4 standard regulates the lower network layers of commercial and custom wireless personal area network (WPAN). It focuses on low-speed, low-cost, ubiquitous communication between devices. The emphasis is on the transmission of nearby devices with little to no underlying infrastructure. The IEEE 802.15.4 PHY and MAC layers are used by many IoT protocols, such as ZigBee and WirelessHART. This standard features real-time reservation of Guaranteed Time Slots (GTS), collision avoidance through CSMA/CA, and integrated support for encrypted communications. Most commercial transceivers also include power management functions such as link quality and RSSI. The standard natively supports time-sensitive applications operating in pure CSMA/CA or TDMA access modes.

2.3.3 LTE - mobile communication

Worldwide there are different generations of mobile communication standards such as second-generation (2G), third-generation (3G), and fourth-generation (4G-including LTE). Moreover, the newest generation (5G) is starting to be used in Europe, and several specific IoT protocols have been designed to support low-power and long range communications. Today, LTE-M and NB-IoT are available for final users. Data rates for these standards range from 9.6 kbps (2G) to 100 Mbps (4G).

2.3.4 Bluetooth

Bluetooth is based on the IEEE 802.15.1 standard. It is a worldwide low power, low-cost wireless communication technology suitable for data transmission between mobile devices over a short-range (8–20 m). It defines a personal area network (PAN) communication operating in the 2.4 GHz band. The data rate spans from 1 Mbps to 24 Mbps. The latest version, BLE 5.0, is the ultra low power version of this standard.

2.3.5 UWB

UWB is relatively recent, and nowadays there is greater adoption of this technology. It features two essential attributes – the prospect of very high data rate communications and the ability to allow very accurate ToF (time of flight) measurements. The very high data rate, 6 Mbps, is supported by several companies under the IEEE 802.15.3a standard. The ability of UWB to provide very accurate location estimation through the precise measurement of ToF has been exploited by a few companies using proprietary systems. The IEEE 802.15.4 (2015) incorporates the UWB PHY into a WPAN standard, allowing, for the first time, an accurate location estimation together with energy-efficient, high data rate communications.

2.3.6 ZigBee

ZigBee (IEEE 802.15.4) defines WPAN specifications with simple devices typically operating in the personal operating space (POS) of 10 m. ZigBee multi-hop, self-organized, encrypted, and reliable mesh networking with at low power consumption, enabling long battery lifetime. The standard defines two different device types: a full-function (FFD) and a reduced-function device (RFD). The full-function device can operate in three modes: a PAN coordinator, a coordinator, or a device. An RFD is specifically designed for extremely simple applications, such as a light switch or a temperature sensor. They may not send large amounts of data; consequently, the RFD can be realized using minimal resources and memory capacity.

2.3.7 6TiSCH

The Internet Engineering Task Force (IETF) standardizes protocols to cover the increasing demand for IP-enabled IoT devices. The IETF IPv6 works over the Time Slotted Channel Hopping (TSCH) mode of the IEEE802.15.4-2015 standard, which supports multi-hop topologies with the IPv6 routing protocol for low power and lossy networks. Moreover, it defines the control plane to match link-layer resources to the routing and application needs. A secure and lightweight join process combines link-layer security features with a joining procedure using the Constrained Application Protocol (CoAP). The tutorial in [21] provides an overview of the 6TSCH suite, including sublayers and a list of the existing open-source and commercial implementations.

2.3.8 Near-Field-Communication

Near-Field-Communication (NFC) is a set of short range (typically requiring a separation of 10 cm or less) communication protocols between two electronic devices. It offers a low-speed connection which can actively generates an RF field that can power a passive target. This enables the creation of very simple and low cost devices, such as unpowered flexible stickes or cards. NFC operates at 13.56 MHz supporting bitrates from 106 kbps to 424 kbps. Its standards cover data exchange formats and communications protocols, and are mainly based on ISO/IEC 14443, ISO/IEC 18092, and FeliCa. The NFC is mainly designed for passive devices, which goes beyond the scope of this disseration, so it is not further discussed.

2.4 Edge computing

Edge computing defines a distributed computing paradigm that keeps computation and storage closer to the source data location to decrease the latency and save bandwidth [59]. The origins of edge computing lie in content delivery networks, created in the late 1990s, to serve web and multimedia content from edge servers deployed close to users. Nowadays, these networks evolved to host applications and application components at the edge sensors, resulting in the first commercial edge services that hosted applications such as shopping carts, dealer locators, real-time data aggregators, and social healthcare monitoring. State of the art edge computing significantly extends the above mentioned approach through virtualization, making it easier to deploy and run a broader range of applications on the edge servers [59].

The increase of IoT devices at the edge of the network produces a massive amount of data to be sent and computed at data centers, pushing the network bandwidth to the limit [60]. Despite the improvements in network protocols, data centers struggle to guarantee acceptable throughput and response time, which is a critical requirement for many applications. On the other hand, devices at the edge continuously consume data from the cloud, forcing companies to build content delivery networks in order to decentralize storage and service provisioning, leveraging physical proximity to the end-user. Similarly, edge computing aims to move the computation away from remote data centers towards the edge of the network, by exploiting mobile phones or low-power smart objects to perform tasks and provide services on behalf of the cloud. It is possible to provide storage, service delivery, content caching, and IoT management by moving intelligence to the edge, resulting in better transfer rates and response times. At the same time, distributing the logic in low power and computationally constrained devices introduces new issues and challenges [60].

The distributed nature of edge computing introduces a transformation in security schemes used in standard cloud computing. Edge nodes may also be resource-constrained devices, limiting the applicability of complex security methods. Hence a shift from a centralized, top-down infrastructure to a decentralized trust model is advisable. On the other hand, by keeping information at the edge, it is possible to move the ownership of data from service providers to end-users, enhancing the whole system's data privacy.

Scalability in a distributed network faces different challenges, such as the heterogeneity of the devices, performance and energy constraints, the highly dynamic deployment condition, and, lastly, the reliability of the connections, compared to the more robust infrastructure of data centers. Moreover, security requirements might introduce additional latency in the communication between nodes, which may slow down the scaling process.

Smart management of the IoT network is crucial in order to maintain a high quality of service. If a single node goes down, end-users should still access a service without interruptions. Moreover, edge computing systems need to provide automatic actions to recover from a failure and, eventually, alerting the operator about the error. To this aim, each device needs to maintain the entire distributed network topology, so that detection of errors and recovery becomes easily applicable.

Thanks to the proximity of the analytical resources to the end-users, sophisticated analytical tools, and Artificial Intelligence (AI) tools can run on the system's edge. This placement at the edge helps to increase operational efficiency and offers many advantages to the final application while reducing the average power consumption by decreasing non-strictly necessary transmissions [27, 28, 57, 61]

We observe an unprecedented boom in AI, particularly deep learning (DL), which has made a remarkable progress in various areas such as autonomous driving, surveillance, computer vision, and health care [62]. To this goal, AI has been broadened from a centralized fashion (i.e., cloud-based) to low power devices in a distributed fashion (i.e., edge-based), opening a new era called edge AI, with dramatic advancements such as substantially changing everyday technology, social behavior, and people's lifestyles. From the definition, edge AI is an extension of edge computing that predominantly focuses on running AI algorithms on low power and constrained devices [62]. It refers to a set of connected machines and systems for sensing, caching, processing, and analysis that have proximity to where data is collected [62]. Its main purposes are to (i) improve the data collection efficiency and computing scalability, (ii) reduce the data processing and transmission latency, (iii) protect user privacy, and (iv) increase information security. Edge AI further enables a wide variety of new promising applications. For example, billions of mobile users are exploiting various smartphone applications such as translation services, food and social life event proposals, and health monitoring services. Another appealing application is the Internet of Things, such as smart home, smart industry, smart city, and the internet of vehicles.

Edge computing is bringing radical changes by enabling decentralized computing capability as close as possible to the data's origin. By leveraging edge computing, the edge AI offers the following advantages. For example, achieving ultra-low latency for time-sensitive tasks with reduced transmission energy cost and bandwidth requirement. With devices generating more data than ever, the cost of bandwidth for large-scale data transmission to the cloud adds up, and often even harder to be sustained. With edge AI, most data processing and computing can be done at the edge, mainly saving the data transmission time and energy from the edge to the cloud. Reduced latency is especially crucial to time-sensitive tasks such as real-time decision making, retailing, and industrial control. As AI has been rapidly advancing, it has been largely constrained by the scalability problem, i.e., the complexity of the AI algorithm design and training grows exponentially with the amount of collected data. Hence, a collaborative machine learning framework without centralized training data is more than ever appropriate. Federated learning enables devices to collaboratively memorize a shared prediction model while keeping all the training data on-board, decoupling the ability to do machine learning from the need to store and process the data in a remote server.

In conjunction with the immense benefits and opportunities, edge AI brings significant challenges with its device constraints and privacy and security concerns. Besides, the rapid advancement of edge AI will drive future trends with corresponding challenges. To empower edge AI, such challenges must be addressed. One of the biggest challenges for edge AI development and deployment are high-performance requirements under strict device constraints. For example, many applications need high accuracy and low latency, such as surveillance and self-driving, requiring real-time performance (e.g., no less than 30 frames per second) and high precision. Similarly, for real-time trading, retailing, and media processing, low latency deployment is essential. On the other hand, most edge devices have strict constraints such as available device resources (computing capability), memory capacity, and power budget (battery endurance). Therefore, developing and deploying edge applications with satisfying performance within strict constraints is a challenge that requires novel design methodologies. These methodologies are classified into the following categories:

1. **Software development:** The development of high-quality ML algorithms, neural network models, to be deployed to edge devices.
2. **Hardware deployment:** The optimization techniques for efficient implementations of software algorithms on the edge devices, aiming at the high quality of service (QoS) such as low latency, high throughput and low power and energy.
3. **Software/hardware co-design:** Co-design methodologies simultaneously develop algorithms and their hardware implementations, to improve the solution quality for both software and hardware designs.
4. **Scalable benchmarking for models, software, and hardware:** Comprehensive benchmarking and profiling provide informative guidance for higher quality software and hardware designs.
5. **Design automation and design space exploration:** Given the large scale and heterogeneous design spaces, design automation is indispensable for improving design optimality with high efficiency.

2.4.1 Approximate computing

It is essential to improve the energy efficiency for edge AI emerging workloads to keep pace with the growth of data that need to be processed. Such algorithms usually feature an intrinsic error-resilience property [63]. They process noisy and redundant data from a plethora of heterogeneous input sources, such as various types of digital sensors (approximate inputs). Moreover, the approximation is enhanced by using algorithms that are frequently stochastic (e.g., iterative algorithms). Furthermore, these applications usually do not require to compute unique or golden numerical results ("acceptable" instead of precise outputs). For example, in multimedia processing, due to humans' limited perceptual capability, occasional errors such as dropping a particular frame or a small image quality loss rarely affect a user's satisfaction. On the other hand, it is frequently energy-efficient to guarantee fault-free calculations as AI technology advances to deep neural networks. This is because nano-meters circuits are increasingly sensitive to parameter variations and faults at an advanced technology node with low supply voltage and ever-increasing integration density [63]. Conventional fault-free computing requires guard-bands and redundancies at various levels, causing significant energy and complexity overhead. Motivated by the above challenges, one promising solution, known as approximate computing, has attracted significant traction from both industry and academia. By relaxing the numerical equivalence between the specification and implementation of error-tolerant applications, approximate computing deliberately introduces "acceptable errors" into the computing process and promises significant energy-efficiency gains [63].

2.4.2 Quantized neural networks

Modern Deep Neural Networks (DNN) usually run on servers, personal computers, or smartphones. Even in the most constrained scenario, their execution can count on GB of RAM and high-end processing capability available under a few watts' power envelope. Conversely, deploying DNNs on an IoT end-node has to deliver comparable performance while dealing with limited computational capabilities, strict constraints in memory (a few MB off-chip, and typically 1MB on-chip at most), and the maximum energy deliverable by the battery. The deployment of algorithms on the IoT demands aggressive software, hardware, and processing co-optimization to exploit these systems' scarce resources [64]. In particular, RAM's limited availability constitutes a fundamental limitation to the maximum performance of an embedded DNN compute system [65]. Recently improvements such as quantized DNN inference [66] aim at reaching a DNN's full precision accuracy and reduce memory footprint while using exclusively 8-bit (or smaller) integer data. On the hardware side, ISA extensions and accelerators [67, 68] have been introduced to speed up the computation, lessen the impact of memory constraints, and minimize energy consumption. Nowadays, 8-bit networks are now supported by most of the NN frameworks, such as TensorFlow and PyTorch.

In a Quantized Neural Network (QNN), all tensors t (e.g., inputs x, outputs y, weights w) are real-value discretized (they assume only a countable set of values), and are represented in a specific range $[\alpha_t, \beta_t)$. On MCUs, it is possible to map these quantized tensors into purely integer variables, namely integer images of the discretized tensor. A detailed description and mathematical background on QNNs is provided in [69], further details and results in this thesis take for granted a basic knowledge of quantized neural networks.

2.4.3 Neural network pruning

Pruning is a technique in deep learning that aims in developing smaller and more efficient neural networks. It is a model optimization technique that involves eliminating unnecessary values in weight tensor. The result is a compressed neural network that runs faster, reducing the computational cost involved in inference. This is crucial when deploying DNN to memory constrained edge devices. Multiple works have been presented in the past years, such as [70, 71].

Chapter 3

Wireless protocols for IoT

This Chapter presents one of the most crucial topics in IoT: wireless sensor network technologies, such as modulations, MAC methodologies, and routing mechanisms. Moreover, the text specifically studies and assesses power consumption, presented as Energy Per Bit (EPB) and/or energy per packet (EPP). The following description and results span between the common IoT standards, such as WiFi, LoRa, LoRaWAN, Bluetooth, BLE, and NB-IoT, also providing a detailed overview of emerging or custom technologies: UWB, ATWR, E-LoRaWAN, S-LoRaWAN, Wake Up Radio (WUR).

In addition to a standalone description, technology comparison is presented, highlighting the main differences in terms of throughput and energy consumption. Design guidelines are also proposed to help engineers to select the most appropriate protocol for each targeted application, even providing an open-source platform that can support many standards. Indeed, it supports WiFi, LoRaWAN, and BLE with parallel and independent execution.

3.1 LoRa and LoRaWAN: Introduction

In past years, academia and industry have dedicated significant effort to develop low-power wide-area networks as a new category of wireless communication standards [72, 73]. Among commercial protocols, LoRaWAN, together with the LoRa modulation developed by Semtech, are gaining popularity [14, 74–81]. The public availability of the specifications [82] combined with low-cost certified transceivers [83] and the fact that a LoRaWAN operates in the unlicensed radio spectrum, meaning that anybody can use the radio protocol without having to pay million dollar fees for transmission rights, motivates the global attention on this communication standard. LoRa offers a link range up to 10 km with a single gateway, which is capable of supporting thousands of different devices; moreover, a unique feature of LoRa is the ability of the gateway to demodulate multiple transmissions simultaneously, with different data rates, on the same channel and time. LoRaWAN specification describes three classes of operations: low-power bi-directional end-devices (Class A), scheduled downlink transmissions (Class B), and Class C designed for always-on bi-directional actuators. This section focuses the attention on the most common used Class A.

Despite the initial popularity, multiple issues remain open about the scalability and the maximum traffic supported in massive LoRaWAN installations. Many LPWAN protocols promise to manage thousands of devices, but the impact of the LoRaWAN parameters on large-scale networks, with packet time that may be $> 1\ s$, is still not well explored. Indeed, in [84, 85], and [86] the scalability of LoRaWAN is studied, and the results in [84] show that only 120 nodes can be managed with a packet transmission every 22 minutes with static configurations. The performance improves growing up to 1600 with dynamic configurations that minimize the time of air. In [76], the authors show an impressive reduction of channel capacity, up to 2×, when the message acknowledgments are required. However, these features are not enough for future IoT deployments [76]. While many existing works have studied LoRaWAN networks' scalability, most of them do not consider the impact of downstream traffic in channel throughput, interference, and issues related to real deployments, such as clock drift and delays produced by restricted computational resources of gateways and servers.

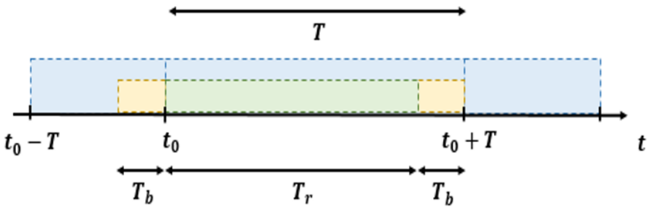
This Section aims at providing an analysis of the channel throughput of the LoRaWAN protocol, and at investigating the issues associated with scalability. The results can provide relevant guidelines for large-scale deployments and physical configurations. A specific network simulator offers information about LoRaWAN Class A with varying number of devices and packet time of air. This work studied the actual payload throughput related to the protocol overhead, which decreases with a factor of 2.22× (in the worst case) compared to Pure-ALOHA (P-ALOHA). After that, a solution to improve network performance is proposed; it is based on Slotted-ALOHA (S-ALOHA). Moreover, a reliable synchronization algorithm was designed, optimized for low-power devices, with an accuracy of 5.37 ms and a success rate of over 99% (Table 3.1). The reference time is used to keep aligned all the end nodes in the network, allowing deployment of Slotted LoRaWAN on a upper layer. In-field simulations and tests performed with 24 end nodes show a throughput improvement of 2×, and 5.8× in crowded conditions. Final results show that the low-power capabilities of Class A are preserved, and the Slotted LoRaWAN can be implemented on existing commercial products.

The number of IoT devices is expected to rise at an annual rate of 32% and market forecasts expect 21 B devices by the end of this decade [73]. Some of the newest applications need to cover huge buildings or entire cities at a very low cost and energy use. These challenging requirements are hard to reach using traditional methods and infrastructures such as WPAN or cellular [72]. Large-scale IoT deployments are becoming a reality, as networks are being installed for urban monitoring applications [73], smart cities [87], and intelligent transportation systems [88]. Several radio standards, such as LoRa, Sigfox, IEEE 802.15.4 NB-IoT, and BLE 5.0, are currently competing in the arena of Device-to-Device (D2D) low-power, long-distance communication [89].

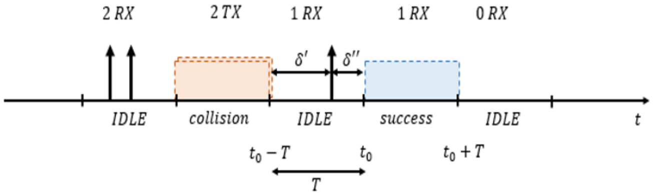
Many communication technologies aimed at low power have been proposed. It is possible, as discussed in [76], to split them into two categories: low power local area networks, such as Bluetooth Low Energy, IEEE 802.15.4, which are usually employed in short-range personal area networks; and low power wide area networks, with a coverage greater than 1000 m. This second category includes Sigfox, LoRaWAN, and DASH7 as major players. Sigfox is a commercial standard which permits remote devices to connect to an access point through Ultra Narrow Band modulation. It is a proprietary technology, thus full specifications are difficult to obtain, so it is tricky to carry out a comprehensive comparison with other protocols. Each end-device can send 140 messages per day, with a payload size of 12 octets, at a data rate of up to 100 bps. Each Sigfox access point can handle up to a million end-devices, with a coverage area of 30–50 km in rural areas and 3–10 km in urban areas [90]. The limit on payload and number of packets makes Sigfox networks hardly usable in application scenarios where communication is not very sporadic: LoRaWAN is less prescriptive and more flexible.

DASH7 [91] is a full Open Systems Interconnection (OSI) stack that operates in unlicensed ISM bands: 433 MHz, 868 MHz, and 915 MHz. It originates from the ISO 18000-7 standard. DASH7 provides communication in the range of 2 km with low latency mobility support, and multi-year battery life. A secure AES 128-bit key encryption is supported with a data rate up to 167 Kbps. However, DASH7 needs more energy per bit than other protocols like LoRaWAN, and for applications that do not require low latency uplink or synchronization, LoRaWAN would be preferred. Moreover, during fast motion, DASH7 is not as reliable as other LPWAN solutions.

In LoRaWAN architecture [20], sensor nodes communicate with the gateway, which serves as a bridge between the nodes and a network server. Three types of functional classes are defined for end-devices: A, B, and C. Currently, several papers analyze LoRa performance [74–76]. In these works, LoRa deployments are compared in terms of network throughput and power consumption. In [79] the coverage of LoRa is studied and in [84], the authors analyze the LoRa network capacity and propose LoRa-Blink to



(a) Slot width definition in S-ALOHA.



(b) Example of free, collided and successful slot.

Figure 3.1

support multi-hop communications. In [92], Casalas and Mir present an analytical model that characterizes the current consumption, lifetime and energy cost in LoRaWAN Class A. The proposed model allows quantifying the impact of physical and MAC layers and bit error rate and collisions on energy performance.

As in the [20] Class A ISO/IEC ISM regulations, the end nodes and gateways can transmit "at will" without any carrier sensing. Thus LoRaWAN's MAC works similar to ALOHA [93]. The ALOHA access mechanism can cause substantial inefficiency in LoRaWAN networks, as highlighted by the simulation study reported in [94], which confirmed that increasing the number of gateways cannot eliminate fast saturation, but in some cases can improve the global performance. Indeed, generating packets following the Poisson law and apply uniform distribution of the payload lengths between one and 51 bytes, as simulated in [76], the maximum theoretical channel capacity usage is 18%, which is comparable with the Pure-ALOHA (P-ALOHA) throughput, configured with a link load of 0.48. In this scenario several packets could be lost; indeed, at the maximum channel load, around 60% of the transmitted packets are dropped because of collisions. To verify the correct reception of each packet, an acknowledgment (ACK) method could be applied. Still, it requires two successive transmissions to be successful, the payload in uplink and ACK in downlink, thus increasing the collision probability with other messages. S-ALOHA protocol is used in local wireless communications [95] since the '70s. The channel time is divided into slots (Fig. 3.1), which have fixed-length T and are composed of two parts: the transmission time (T_r) and the tolerance interval (T_b), as presented in Fig. 3.1(a). Every end-node must transmit a packet only at the beginning of a slot. If two or more end nodes send their packet within the same fixed-length period, a collision occurs; otherwise, no overlap is generated, and the data are correctly sent (Fig. 3.1(b)). If not a single terminal accesses in one slot, it will be free, and no device will start the transmission in the middle of the time-slot. In S-ALOHA, the maximum theoretical channel throughput is 37%.

3.1.1 LoRa physical layer and LoRaWAN

LoRa is a chirp spread spectrum modulation, in which the frequency increases or decreases with time. It is originally developed for radar applications; indeed, the spread spectrum modulation has a constant amplitude and uses the whole bandwidth, in a linear or non-linear way, from one end to another end, in a defined time. The maximum frequency delta allowed between the receiver (gateway) and transmitters (end nodes) can reach 20% of the bandwidth without impacting decoding performance [76]. This feature is useful to design cheap devices with low-quality crystals, such as commercial products with 80 $ppm/°C$. A LoRa typical transceiver, such as the SX1276 [76], provides a link budget up to 168 dB with a sensitivity of -148 dBm at the receiver side. Moreover, since the symbol period is larger than a typical noise spike, interferences are easily filtered through error correction codes. Code Rate (CR), Bandwidth (BW), and Spreading Factor (SF) affect the LoRa modulation; indeed, a single LoRa symbol is composed of two SF chirps, with a frequency range that covers the entire band. The chirp symbol rate is directly dependent on the bandwidth, which generates different consequences as specified in [76]: (*i*) increasing the SF, the chirp frequency span is divided by two; (*ii*) the period of each symbol is multiplied by two; (*iii*) since more than one bits are transmitted in a symbol, the bit rate is not decremented with a ratio of two; (*iv*) the bit rate at given SF is proportional to the bandwidth, doubling the bandwidth will double the transmission rate. Over LoRa modulation, forward error correction is used to enhance communication reliability. This method is implemented using four different coding rate (CR) approaches, denoted CR1 to CR4. In addiction, a drift correction mechanism is applied to increase robustness to frequency variation over the LoRa message's timescale. Eq. 3.1 gives an estimation of the equivalent bit rate (EBR), while the duration of uplink and downlink transmissions depends on LoRa's parameters[1] such as SF, BW, CR that can be expressed as the sum of the time needed to transmit the preamble and the physical message (Eq. 3.2).

$$EBR = SF\left(\frac{BW}{2^{SF}}\right)CR, \tag{3.1}$$

$$T_{tx} = T_{preamble} + T_{PHYMessage} \tag{3.2}$$

---

[1] The LoRa calculator is available online at `http://www.rfwireless-world.com/calculators/LoRa-Data-Rate-Calculator.html`; LoRaWAN stack is available online at `https://www.st.com/en/wireless-connectivity/lorawan-technology.html`

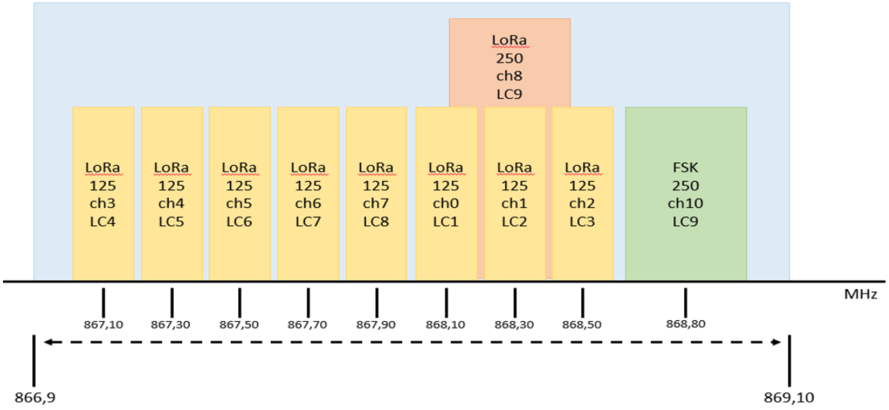


Figure 3.2: Overview of LoRaWAN channels.

Equations 3.3 and 3.4 represent how these two terms have been calculated, where $N_{preamble}$ is the number of preamble symbols and N_{PHY} indicates the number of symbols transmitted in the physical message. They can be determined as shown in Eq. 3.6. T_{sym} (Eq. 3.5) is the duration (in seconds) of a symbol and depends on SF and BW.

$$T_{preamble} = T_{sym}\left(N_{preamble} + 4.25\right), \tag{3.3}$$

$$T_{PHYMessage} = T_{sym} \cdot N_{PHY}, \tag{3.4}$$

$$T_{sym} = \frac{2^{SF}}{BW}, \tag{3.5}$$

$$N_{PHY} = 8 + max\left[ceil\left[\frac{28 + 8 \cdot PL + 16 \cdot CRC - 4 \cdot SF}{4(SF - 2 \cdot DE)}\right] \cdot (CR + 4), 0\right]. \tag{3.6}$$

In Eq. 3.6, PL (Payload Length) denotes the number of bytes in the physical payload, CRC indicates the presence (value 1) or not (value 0) of the CRC field in the physical message and DE indicates if the clock drift mechanism of the crystal reference oscillator is used (value 1 for SF12 and SF11, 0 for others). The transmission data rate can be obtained as shown in Eq. 3.7:

$$DR = SF \cdot \frac{BW}{2^{SF}} \cdot \frac{4}{CR + 4}. \tag{3.7}$$

The radio channels used in LoRa depend on the country rules. In this work, all the tests are performed in Italy, Europe, where the ISM (Industrial Scientific Medical) band is at 863-870 MHz. LoRaWAN provides 9 different channels. Channels from 0 to 9 provide a bandwidth of 125 kHz and must support data rates between 0.3 kbps to 5 kbps; furthermore, channel ten is allocated for FSK modulation with a bandwidth of 250 kHz. Fig. 3.2 presents the frequency characterization of each LoRaWAN channel. A relevant feature of LoRa modulation is that different SFs are orthogonal, this means that a LoRaWAN gateway can receive a multiple transmission on different SFs simultaneously and can apply this methodology for each channel. Due to local rules imposed by ISO/IEC

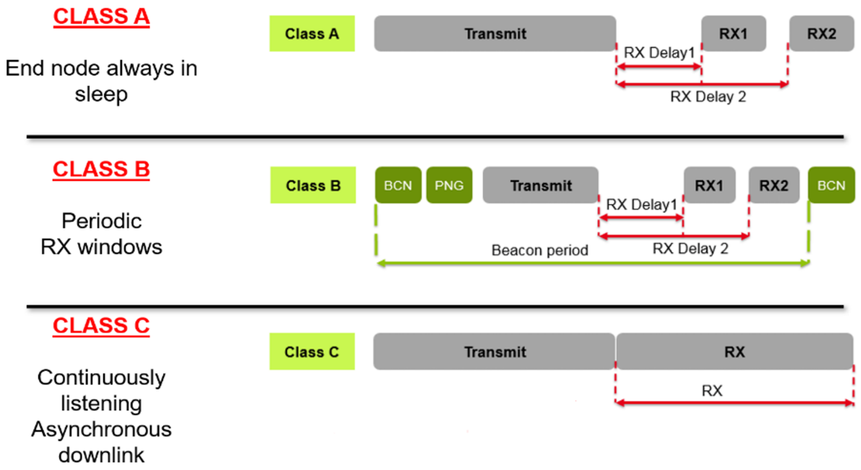


Figure 3.3: Uplink and Downlink methodology in Class A, B, and C.

ISM regulations, wireless sensor devices working on ALOHA MAC access cannot occupy more than 1% of the channel time. It is essential to underline that, as long as each limitation is respected, each end device can transmit on different channels and in different sub-bands to increase its overall throughput [80].

Class A is focused on the end node side; indeed, all transmissions are scheduled by the end node, whereas the server can transmit only in one of two receive windows opened after the previous uplink transmission, as presented in Fig. 3.3. Therefore, any packet that the application level needs to transmit must wait until the next scheduled receive windows. In the MAC layer, the typical packet is composed by a preamble, a physical header, a variable payload length, and error detection bytes (CRC). The CRC is present only in the uplink message; thus, the downlink data is mainly used for ACKs. LoRaWAN defines different strategies for the downlink channel and DR selection in RX1 and RX2 windows; in this Section, LoRaWAN is configured to use the same DR and channel for uplink and downlink transmissions; moreover, RX2 window is disabled to evaluate the network performances in static conditions. Indeed, with the RX2 enabled, the downlink period could change up to 200%. The minimum time width for RX1 and RX2 must guarantee at least the time required to effectively detect a downlink preamble [20], and in case it is detected, the transceiver must remain active until the reception of all data.

3.2 LoRaWAN MAC Model

Class A is designed for battery-powered devices targetting a very long expected lifetime. It supports bi-directional communications, but downlink transmissions are constrained by two synchronous short intervals after each uplink to the gateway. Class

B and Class C are designed for end nodes with no critical energy constraints; thus, they allow for more receiving slots at specific scheduled times, and synchronized beacons are sent from the gateway. This section focuses on an S-ALOHA scheme for LoRaWAN battery-powered sensor nodes.

Defining S as the average number of packets generated per transmission time; the traffic source λ consists of a large number of users who collectively form an independent Poisson source with an aggregate mean generation rate of X packets/s, the packet time width is supposedly fixed with a period of T seconds (Eq. 3.8). Moreover, each user generates packets infrequently and each of those packets can be successfully transmitted in a slot smaller than the average time between successive packets [96]. S is also expressed as the channel throughput rate. Each user delays the transmission of a previously collided packet by some random time, chosen, for example, uniformly between 0 and T_{max}. Therefore, the traffic injected into the channel consists not only of new packets but also of previously collided packets: this increases the mean traffic generated, usually denoted with G (Eq. 3.9).

$$S = \lambda T, \tag{3.8}$$

$$G \geq S, \tag{3.9}$$

$$G(n) = \lambda(n)T \tag{3.10}$$

$$S = G(n) \cdot P_{succ} = \lambda(n)T \cdot e^{-\lambda(n)2T} \tag{3.11}$$

In P-ALOHA, a single broadcast is successfully performed if the channel remains free during the time period 2T (vulnerability period). The probability that there are no transmissions in the 2T period is P_{suc} [96]. The total channel traffic could be expressed as presented in Eq. 3.10. With these hypotheses, it is possible to obtain a maximum channel throughput of 18%.

This model approximates LoRaWAN Class A in the unacknowledged configuration, when there is an uplink transmission without ACKs from gateway(s). To model the single channel throughput in the half-duplex mode, this section presents a statistical model based on the network simulator, statistical analysis, and measurements from a real LoRaWAN testbed.

As shown in Fig.3.3, in LoRaWAN Class A there is a one-second delay between transmission and RX1 windows; this period could be used by other end nodes to transfer data, but the probability of successful communication in the RX Delay 1 is very low since LoRa modulation generates a remarkable packet's time of air. With an SF12[2], a BW 125 kHz, CR4, and 25 bytes of payload, the overall time of air is 1253 ms; with these settings, the corresponding ACK time width is 530 ms. With the proposed param-

---

[2] It is the equivalent term of SF = 12, or CR = 4

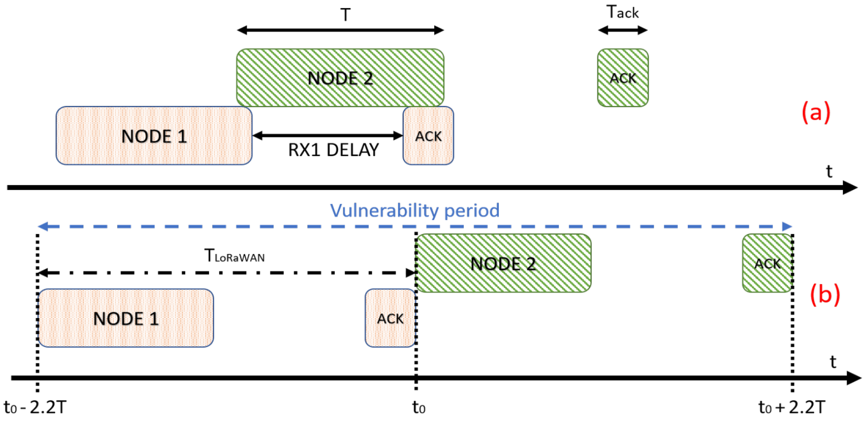
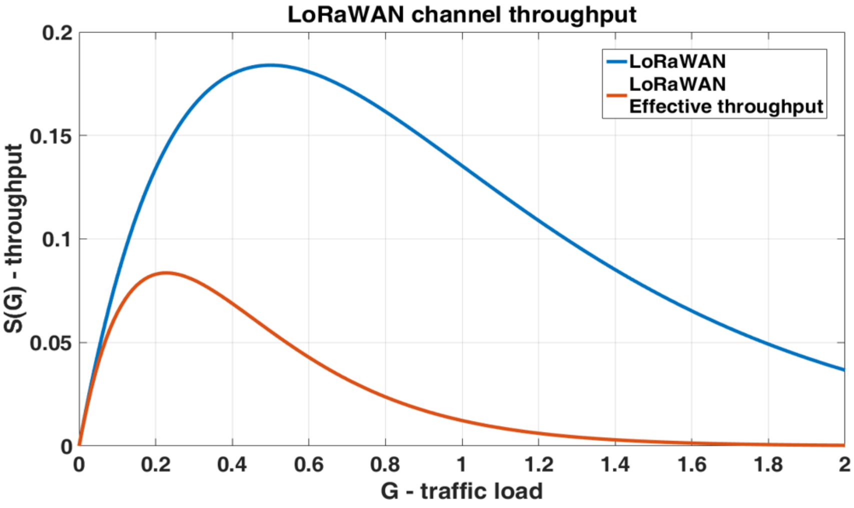


Figure 3.4: Vulnerability period in LoRaWAN with ACK. (a) T is greater than RX1 Delay, a collision must occur; (b) vulnerability period needed to perform a successful link between an end device and the gateway.

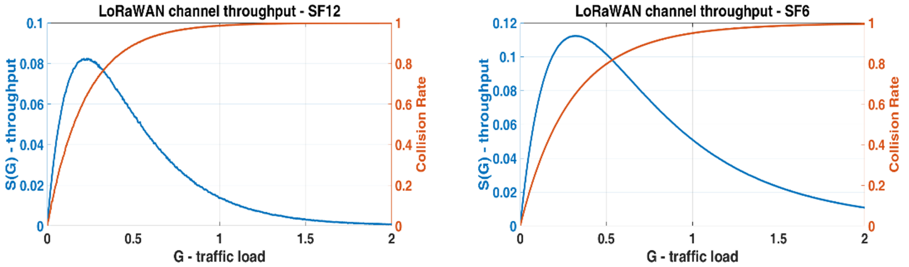
eters, a transmission within the RX1 Delay is not allowed, as shown in Fig. 3.4, and the probability of collision is equal to one. In this study, the overall channel occupation time for each transmission is composed by the time of air of both uplink (T) and downlink (T_{ack}) windows plus the RX1 Delay. Considering T the channel time used to transfer the payload, it is possible to define $2.22 \cdot T$ the total channel time used for each sensor, considering the LoRaWAN overhead; this value is equal to $T_{LoRaWAN}$. Therefore, the ratio between the injected traffic and transmission successfully performed can be expressed as presented in Eq. 3.12, where the vulnerability period scales up from $2T$ to $2 \cdot (2.22 \cdot T)$.

$$S = G(n) \cdot P_{succ} = \lambda(n)T \cdot e^{-\lambda(n) \cdot 2 \cdot (2.22 \cdot T)} \tag{3.12}$$

The effective LoRaWAN throughput decreases drastically concerning the standard P-ALOHA, reaching a maximum of 8% at G equal to 0.25. This outcome shows that the LoRaWAN throughput (for each channel) is inadequate when half-duplex communication is required. To confirm this conclusion, a simulator was developed in MATLAB to evaluate the channel collision rate with different time-of-air configurations. In detail, the simulator generates traffic using a variable number of nodes, and each one works following a Poisson uniform distribution. When a simulation starts, the number of nodes grows from one and stops as soon as the desired (G) is achieved. The maximum is 1000, and each element tries to access the channel with a variable duty cycle and a maximum of 40 attempts. The simulator checks every millisecond if two end devices try to communicate simultaneously, with an overall simulation time of two hours. Fig. 3.5(a) shows the S(G) for unconfirmed and confirmed LoRaWAN single-channel link; indeed, in Fig. 3.5(b) and Fig. 3.5(c) the results of two different simulations are presented. In Fig. 3.5(b) the packet time is greater than the RX1 Delay, in Fig. 3.5(c) the time of air is $\ll$ 1 second, so the probability for a successfull transmission is not equal to zero, and the network performance slightly increases. Fig. 3.5(b) and Fig. 3.5(c) confirm that the proposed model is valid, pointing out that the effective channel throughput in LoRaWAN is tightly constrained. In S-ALOHA, all the components of a LoRaWAN link can be fitted into a single slot (Fig. 3.1(a)), uplink and downlink windows as well as the RX1 Delay, keeping the throughput of the proposed Slotted LoRaWAN (S-LoRaWAN) equal to standard S-ALOHA. As expressed for the P-ALOHA, considering T the channel time used to transfer the payload, it is possible to define $2.22 \cdot T$ the total slot time used, also considering the LoRaWAN overhead. Following these considerations, the maximum channel throughput in S-ALOHA is 16%, doubled with respect to the classic half-duplex LoRaWAN.



(a) Unconfirmed and confirmed LoRaWAN single channel throughput.



(b) Confirmed LoRaWAN throughput and collision rate for one channel acquired by MATLAB simulation. SF12, BW 125 KHz, CR4 Maximum throughput is 8%.

(c) Confirmed LoRaWAN throughput and collision rate for one channel acquired by MATLAB simulation. SF6, BW 125 KHz, CR4 Maximum throughput is 11%.

Figure 3.5: Unconfirmed and confirmed LoRaWAN single channel throughput

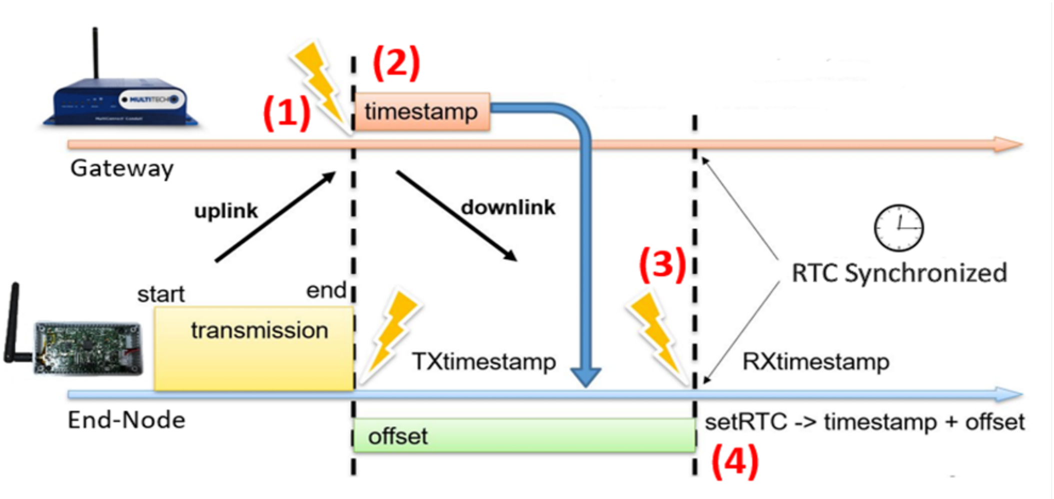


Figure 3.6: Synchronization procedure.

3.3 Real Time Clock Synchronization over LoRaWAN stack

Clock synchronization across devices is fundamental to define the slots used by S-ALOHA. This section makes use of a lightweight synchronization methodology, presented in [97] and successfully tested on an STM32L4 MCU. To generate a time reference for the S-ALOHA, where the slots must be kept aligned in all the devices, the MCU uses a real-time clock (RTC) with an inexpensive external crystal at 32.768 kHz, characterized by a thermal variability between 20 to 80 $ppm/°C$. Since this component drifts significantly with temperature, the MCU needs to re-synchronize its clock frequently. The synchronization scheme is piggybacked on the predefined answer (ACK) of a Class A RX1 window, opened with a maximum error of ± 20 μs delay [97]. Common events for both devices are used as a shared reference for the synchronization algorithm. In a typical half-duplex connection, the node and the gateway must know when to open the RX1 window after a successful uplink. A timestamp at the end of such transmission is saved on both devices to open the RX1 window simultaneously; thus, this event (Fig. 3.6) is used as a reference for the clock synchronization procedure [97]. The synchronization algorithm is based on a clock reference distribution system: basically, at the end of every complete uplink transmission, each node updates its real-time clock. The implemented algorithm works as follows:

1. gateway and node save a timestamp of the moment when the uplink transmission has ended;

2. the node receives, in the first available RX window, then the timestamp is piggybacked in the acknowledgment message;
3. due to the non-negligible flight times and the RX window delay, the node must consider the offset between the timestamp when the uplink transmission has ended (TX timestamp) and when it receives the acknowledgment message (RX timestamp). The node calculates the difference between these two timestamps and adds it to the reference received by the gateway;
4. the code updates its real-time clock with the new timestamp; in this way both devices are synchronized.

The developed framework needs to automatically update the end node RTC to provide a sufficient alignment with respect the reference clock. Moreover, the whole protocol must be compliant with LoRaWAN specification without adding a significant overhead on the system. The proposed approach uses the uplink packet and the RX window already scheduled after 1 second. Indeed, the timestamp saved on the gateway, received at the very end of the uplink transmission, can be sent back with the ACK packet in the first reception window. Hence the end node has the time reference needed for clock synchronization. The overall overhead is very low, only 8 bytes of timestamp, with good scalability and configurability [97]. In many applications the overhead is negligible; for example when multiple transmissions per day are needed, the synchronization procedure comes for free adding the gateway timestamp at some downlink packet. In systems where the end node is meant to send few packets every day, some additional synchronization phases can be needed to ensure a sufficient clocks alignment. Maximum tolerable clock drift depends on three factors: (*i*) maximum uplink time of air; (*ii*) daily duty cycle of each end node; (*iii*) maximum throughput required for the specific application.

On the end node side, the goal of the synchronization firmware is to update the local clock using the timestamp received from the gateway. This process requires two steps: the offset calculation and the sub-seconds management. Since the ACK in response to the uplink takes a variable delay, the offset calculation is needed to update the received timestamp. The offset calculation procedure consists in measuring the time delta between the response packet and the initial instant when the transmission on the end node ends; this tag coincides also with the packet reception on the gateway, which is the moment where the gateway "marks" the message with the timestamp included in the RX1. The electromagnetic propagation speed is considered as a negligible factor respect to the times involved.

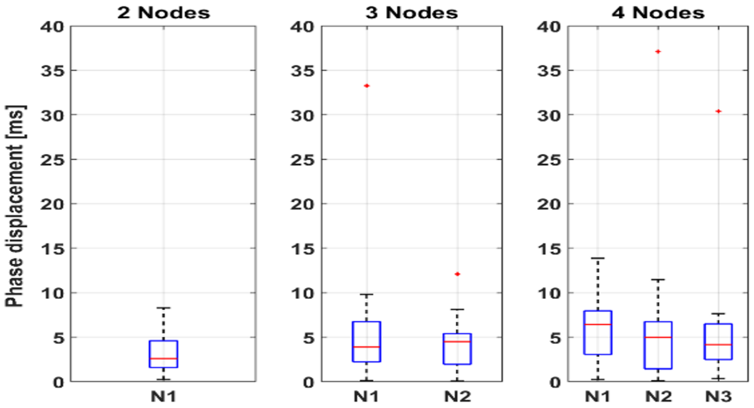
A total of 30 synchronization tests were conducted with 2, 3, and 4 nodes: the results are shown in Table 3.1, where the time difference in milliseconds is expressed taking as reference the end node N0. This evaluation aims to verify the synchronization performance in a network composed of many devices. Indeed, they are programmed to send a request every 15 seconds, which is comparable with 320 end nodes generating a request-rate of 80 minutes each. From Table 3.1, it is possible to notice how the increase of devices significantly reduces the overall accuracy. This is probably due to the management of processes at the gateway operating system. Since the software working on mLinux[3] controls the timestamps, the operation priority on the gateway is not well known. Consequently, the final accuracy associated with the received packet cannot be under one millisecond. Achieved results show that the mean error among all the devices is 5.37 ms, with a maximum of 37.08 ms; such uncertainty must be considered in S-ALOHA deployment to avoid transmission overlaps between adjacent slots. To decrease the error, some LoRaWAN gateways embed a GPS module, used by the physical layer to exactly associate the time reference at the end of the packet frame, but, since this feature is not mandatory, Slotted LoRaWAN works without this support. Fig. 3.7(a) presents the plots of synchronization deviation among end nodes, from two to four, where are indicated the median, and the 25th and 75th percentiles. Fig. 3.7(a) shows that, as the number of transmitter increases, the mean and the width of the interquartile gap (blue box) rises, which is a dispersion index providing an evaluation of how far the synchronization uncertainly moves away from a central value, such as standard deviation. As a result of the increase in the number of devices, the number of "off-average" synchronizations, indicated in the plot with red crosses, also increases. Fig. 3.7(b) represents the maximum synchronization uncertainty between nodes since last clock alignment and provides essential information to calculate the T_b period and the RTC refresh rate, which are dependent on application and transmission period. For example, in the worst case, when the end node integrates an 80 ppm crystal, an error of 200 ms is generated every 40 minutes.

Table 3.1: Results of the measurements carried out to verify the RTC synchronization between multiple end nodes connected to a single gateway.

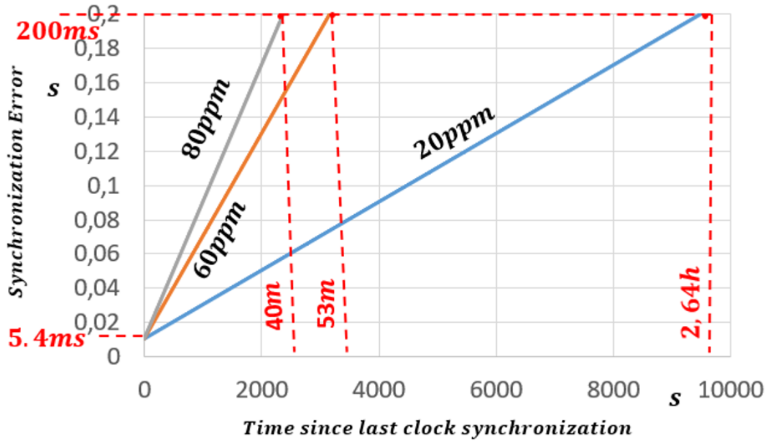
Number of end nodes	Min. synch. error [ms]	Max. synch. error [ms]	Avg. synch. error [ms]	Delta Avg. error [ms]
2	0.072	8.296	3.310	-
3	0.120	33.260	4.705	+42
4	0.243	37.080	5.370	+62

---

[3] `http://www.multitech.net/developer/software/`



(a) Median, 25th, and 75th of synchronization error between 2-4 end nodes.



(b) Typical crystal drift over time.

Figure 3.7

3.4 S-ALOHA implementation over LoRaWAN

Once implemented, at the application level, a lightweight and accurate synchronization system that guarantees acceptable uncertainties of milliseconds order, it is necessary to introduce specific functions for the transmission management on top of the LoRaWAN MAC level. The application layer schedules the uplink transmission only at the beginning of each slot, starting a transfer request only when the internal RTC triggers the event. For calculating the time T_r, it is necessary to consider that LoRaWAN supports different types of configurations depending on the bandwidth, spreading factor, coding rate, and, above all, depending on the payload length of the packet sent. The configuration used in the following chapter is made up of these parameters: SF=8, BW=125 kHz, Preamble=8 bytes, CR=1, and Payload=200 bytes. This is a standard setting in WSN, providing an optimal trade-off between coverage and bitrate. Following [97], we can estimate the uplink time of air, equal to 553.47 ms. It means that considering the 8 bytes added into ACK needed for the synchronization, the lower bound of the slot duration is 1.615 seconds (T_r), which is equal to $T_{LoRaWAN}$ portrayed in Fig. 3.4. The packet size is considerable and close to the maximum allowed of 255 bytes, maintaining an acceptable ratio between the transfer time (T_r) and the overhead time (T_b). The minimal slot duration has been calculated considering that the response in the first window RX1 occurs with the same transmission parameters of the end node. Moreover, the transmission of the ACK message cannot collide with other communications being completely included within the time of a slot. The calculation of the T_b time, the tolerance interval, considers the average alignment error generated by the synchronization algorithm and the clock drift introduced by the quartz oscillator. Considering a crystal with a frequency deviation of 80 $ppm/°C$ (worst case), a phase shift of 200 ms is generated every 40 minutes, a considerable and not negligible value in the case of devices transmit a few times in a day. For calculating T_b, a base-time of 5.4 ms is considered given by the maximum alignment shift reached during the synchronization tests. Considering the clock drift, the time margin is approximately 400 ms. It aims to have an interval of at least 80 minutes between one synchronization and another. Lastly, a T_b of about 385 ms has been chosen, then a slot time T with a total duration of 2 seconds, with a T_b / T_r ratio equal to 25%.

In addition to the slot time definition, to implement the S-ALOHA protocol and make evaluations about the performances and improvements introduced, it was decided to force the nodes transmission on a single channel (channel 6). This is to avoid the random selection of the transmission channel, which would involve the evaluation of the performance on several channels, which goes beyond the scope of this study.

The S-ALOHA protocol must confirm the receipt. In case of packet loss, the end node needs to schedule retransmission in a successive random slot (random backoff). The backoff algorithm works above the LoRaWAN application layer, which calculates the wait time before retransmitting as a multiple of the single slot time. A random natural number is calculated within the interval that can be set by the user [0, NSLOTS]. The randomness in choosing the slot to perform the retransmission is a fundamental feature to avoid devices entering pathological repeated interference corner cases. S-LoRaWAN works on the application layer following a mechanism detailed in the block diagram presented in Fig. 3.8, using two timers and the LoRa transmission libraries. The S-ALOHA state machine has five states: (*i*) Init: in this state, timers used during firmware execution are initialized. There are three timers: *TxLoraSlotted* for the calculation of the time to wait to transmit in the first available slot; *TimerBackOff* for the choice of a random slot for retransmission, and *TimerAck* at the expiration of which packet ACK receipt is checked. Following the initializations, the S-LoRaWAN switches into the Wait state, in which a transmission request is expected. In this state, the MCU

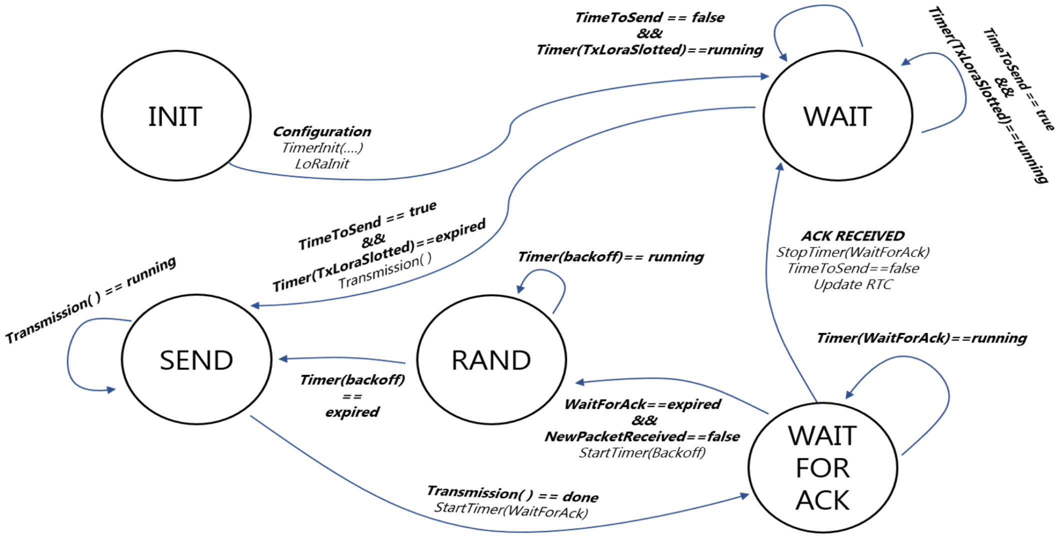


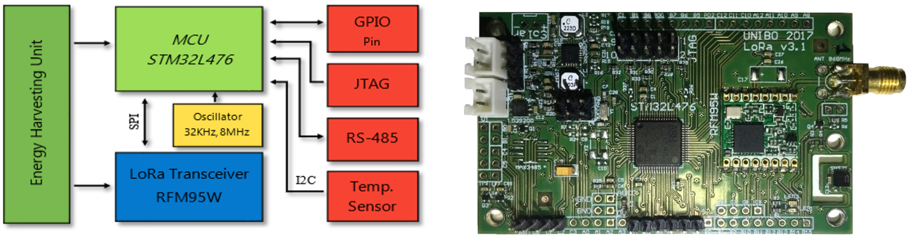
Figure 3.8: Block diagram of the S-LoRaWAN decision states.

is in sleep. (*ii*) Wait: *TimeToSend* variable is periodically checked; when it assumes logical value true, a new packet is sent. The state transition between Wait and Send occurs triggering the *TxLoraSlotted* timer in correspondence of the first time slot in which a transmission is allowed (as foreseen by the Slotted Aloha protocol). (*iii*) Send: in this state, the transmission request is executed, which is responsible for sending the data and calling the *PrepareTxFrame* and *SendFrame* functions. The first deals with preparing the payload, and the latter executes the physical communication with the LoRa transceiver. After that, the system automatically transits to the WaitForAck state. (*iv*) WaitForAck: the system stays in this state until the timer *TimerAck* is triggered or the acknowledgment is received. If the ACK is detected, the system returns to the Wait state until the next transmission request; otherwise, the system switches to the Rand state where the backoff algorithm is executed. (*v*) Rand: as a result of a non-acknowledgment event, the MCU programs the transmission in a randomly selected slot. The *TimerBackOff* is programmed and the device remains in the Rand state until it expires. When it expires, the system automatically transits to the Send state, where the message is retransmitted.

3.5 S-ALOHA evaluation in a real deployment

3.5.1 The end device

The end node used in this evaluation is based on a custom board developed for multiple purposes. It embeds an STMicroelectronics STM32L476, the LoRa RFM95W



(a) Hardware block diagram.

(b) Picture of the custom board.

Figure 3.9: The end device hardware

transceiver, an energy harvester sub-circuit (BQ25570) that manages the power supply and a temperature and humidity sensor (SHT21). Lastly, an expansion connector enables the board to be connected with several analogs and digital external sensors. The current in sleep mode is 4 µA at 3 V with the RTC enabled; the STM32L476 uses 8.25 mA in RUN mode @ 48 MHz. The RFM95W power consumption in TX and RX is respectively 76 (@ 10 dBm) and 11.5 mA. The LoRaWAN firmware comes from I-CUBE-LRWAN libraries package from STMicroelectronics, configured to support S-LoRaWAN. With the MCU programmed to send a packet every 30 seconds, the battery (1000 mAh) lifespan is seven months. Fig. 3.9 presents a picture of the board and the related block diagram.

3.5.2 The gateway

The selected LoRaWAN gateway is a MultiConnect Conduit device. It is a configurable gateway for industrial IoT applications. LTE, 3G, and 2G, plus Ethernet, are available to deploy network connections and data management. The LoRaWAN radio module embeds the Semtech SX1301 and two SX1257, which demodulate the packets received simultaneously on all channels and SFs. In this deployment, the gateway is connected through Ethernet in LAN with the server, and the application software runs on top of Node-Red[4].

3.5.3 The evaluation environment

The custom sensor node was tested in different deployments and applications. The results presented in this study are acquired placing the network indoors, allowing an easy setup and configuration management for all the tests given in the following chapters. The S-LoRaWAN was tested in CINECA (Fig. 3.10), the Tier0 supercomputing center

---

[4] Conduit AEP: LoRa Communication and Node-RED, Available online (2019): http://www.multitech.net/developer/software/lora/conduit-aep-lora-communication/

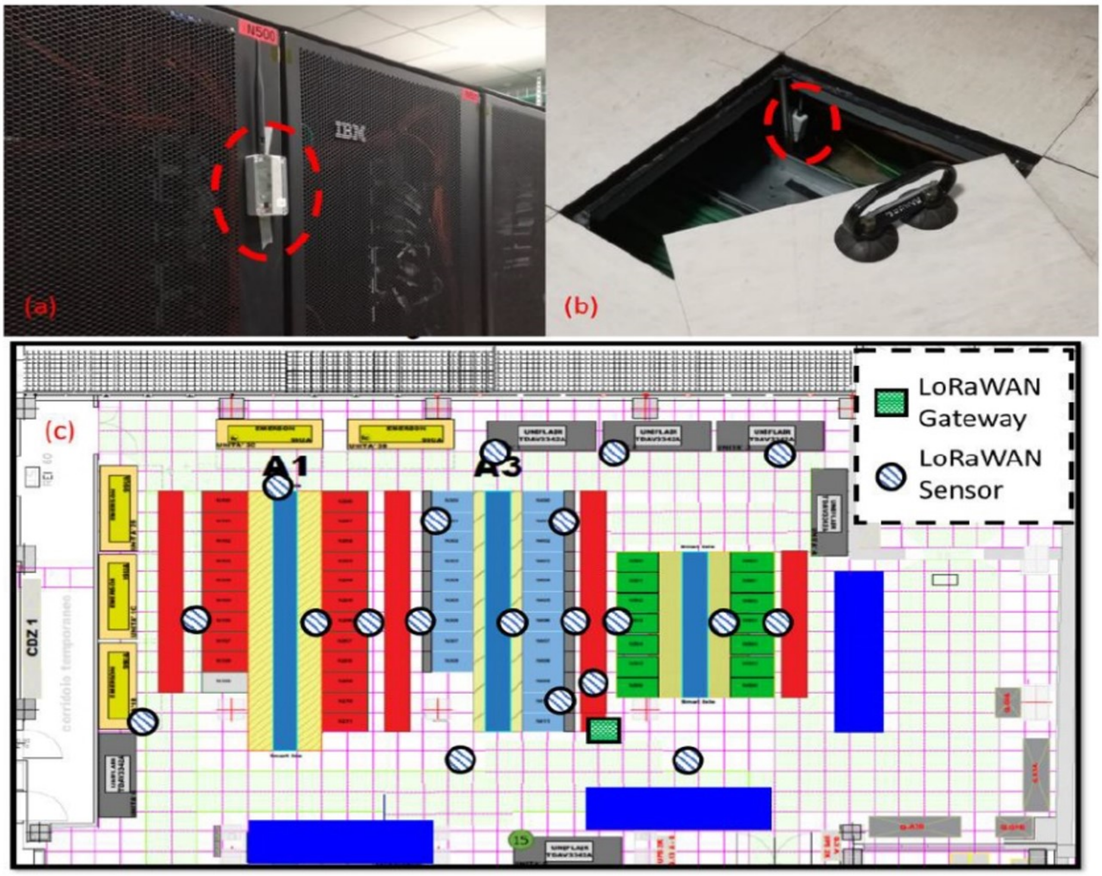


Figure 3.10: Sensor node deployment in CINECA. (a) hallway positioning, (b) the sensor node is under the data center floor, (c) CINECA data center map.

for scientific research in Italy, where the end-nodes were used to monitor in a distributed fashion the temperature evolution of the computer air conditioners to improve the cooling efficiency. In this application scenario, [97] demonstrates a packet collision reduction of 3.4× in comparison with a standard LoRaWAN network with 20 end-nodes. Moreover, the S-ALOHA was included in a structural monitoring application composed by a custom sensor board designed to measure the crack width, see Section 5.1. This application requires hundreds of sensors, which generate thousands of bytes per day. This traffic can overcrowd a standard LoRaWAN network; indeed, the S-ALOHA enables the possibility to double the throughput increasing the overall number of sensors.

3.5.4 S-ALOHA evaluation

To hold a performance analysis of the S-ALOHA and to compare with the standard LoRaWAN, a detailed log of each device status is collected in a database, which is managed by the LoRaWAN server. Two packet-send and packet-loss counters were inserted in the state machine, indicating the total number of packets that each node sent and those lost. The packet-send counter is incremented every time the MCU is sending. On the other hand, to check whether the acknowledgment has been received, a logical value is set to true. Moreover, the gateway detects each channel access time, measuring the effective RTC synchronization of each end node. The payload contains the log information, allowing to read in real-time the statistics of each node.

Tests have been carried out to assess whether the introduction of the S-ALOHA leads to an improvement in the network overall performance. The LoRaWAN protocol is similar in terms of channel access and transmission algorithm to the P-ALOHA. Still, it is necessary to consider that the statistical analysis of the P-ALOHA does not consider the channel occupation for acknowledgment messages. The transmission of the ACK by the gateway in the same channel can, in fact, collide with the other frames that have been transmitted by other devices.

Three tests for the S-ALOHA protocol were carried out in three different network traffic conditions to evaluate the performance gap between the theory, statistical models, and real deployments. The key variables of these tests are the number of devices and the transmission period of the messages. Indeed, other parameters, such as SF and payload size, are kept constant. In all tests, confirmed messages are implemented, and the S-ALOHA protocol, with a back-off algorithm, is implemented in all the devices. Table 3.2 and Fig. 3.11 show the comparison between the protocol performances in terms of traffic offered and disposed of by the network. These values are normalized, so the results are independent of the experiment duration, expressing the overall usage in percentage value. The total number of packets sent in the time unit (packet/second) was considered for the channel traffic calculation (G); this means that, on average, values greater than G = 1 denote that multiple nodes transmit simultaneously in the same slots. The overall throughput from the channel is therefore given by G for the average percentage of packages successfully disposed of, as described in Eq. 3.13 and 3.14. The

Table 3.2: Summary of the results obtained from the tests with Slotted LoRaWAN.

	Test 1	Test 2	Test 3
Test duration [s]	10273	6977	3524
Packets sent	2708	3285	2894
N of end nodes	10	18	24
TX period [s]	15	15	13
TX success rate [%]	69	55	43
Traffic generated (G)	0.264	0.471	0.773
Channel throughput (S)	0.184	0.275	0.332

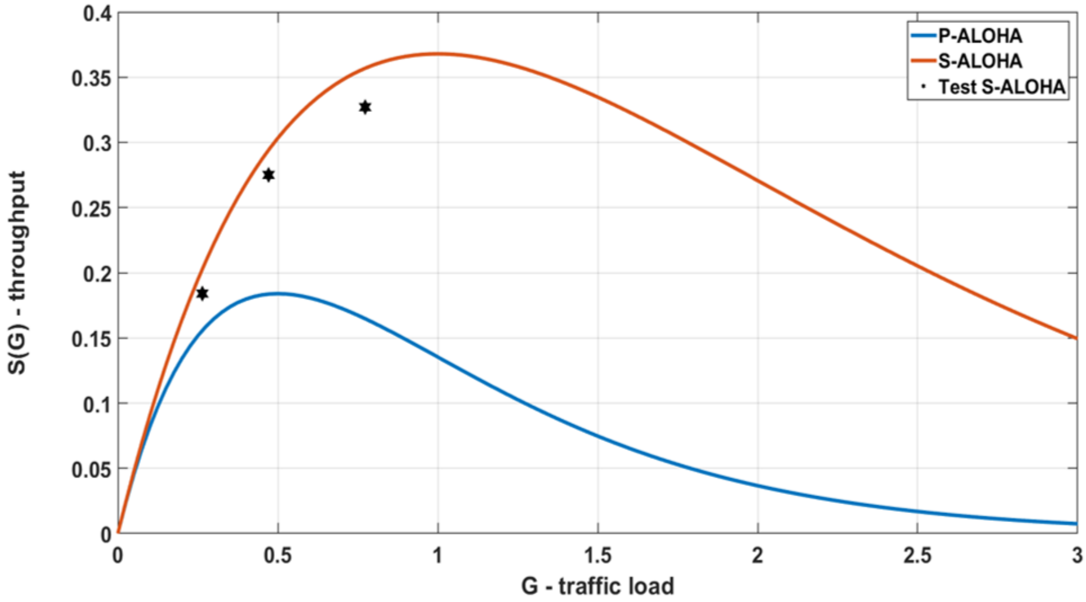


Figure 3.11: Slotted LoRaWAN results.

packet time is calculated considering the $T_{LoRaWAN}$.

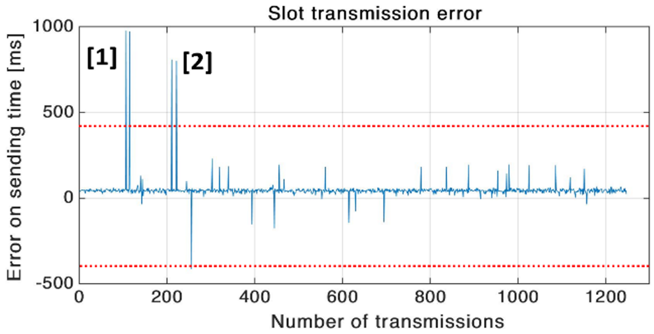
$$G = \frac{Packet_{sent}}{T_{test}}, \tag{3.13}$$

$$S = G \cdot P_{success} \tag{3.14}$$

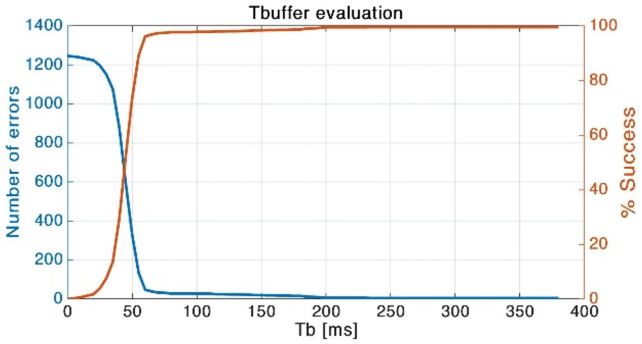
Values reported in Table 3.2 and Fig. 3.11 confirm a correct implementation of the S-ALOHA over a LoRaWAN protocol; indeed, the maximum theoretical throughput is doubled, with a trend similar to S-ALOHA.

3.5.5 Synchronization analysis in Test 3

An in-depth analysis of the synchronization among end nodes is presented here. Test 3 of Table 3.2 is used as the reference; indeed, the high generated traffic and the high packet collision rate on the channel stresses the synchronization procedure. The error is calculated between t_0 (Fig. 3.1(a)) and the packet time of arrival measured by the gateway. With the T_b proposed in Test 3, the overall number of failed slots for all 24 nodes is five, reaching a success rate of 99.6%. In this case, an overlap happens among adjacent slots because the tolerance interval is exceeded. As presented in Fig. 3.12(a), in points (1) and (2), two consecutive slot overlaps are shown. Both are generated from the same device. This issue comes from a reboot; indeed, when a firmware error is detected, the MCU is restarted and the internal RTC needs to be configured again. Therefore, the overall synchronization failures are three. The previous trials are executed with a tolerance interval (T_b) calculated to fulfill the RTC drift in the worst case. The S-LoRaWAN throughput is evaluated in this condition to compare both protocols in a



(a) Synchronization error in ms respect to the beginning of each slot. The maximum error allowed is 400 ms, which is exceeded 5 times. Two overflows are generated from end node reboot due to firmware issues.



(b) Number of slots overlapped as a function of tolerance interval length.

Figure 3.12

typical low power application. However, in Test 3 the T_b could be decreased ensuring a slot success rate over 90%; Fig. 3.12(b) and Table 3.3 show the relationship between T_b and the number of errors. For example, with a T_b of 100 ms, the error rate is practically the same as Test 3, and with 50 ms, the error rate is still acceptable. The tolerance interval length affects the channel throughput; indeed, reducing T_b decreases the time overhead and allows more slots in a unit time.

Table 3.3: Success rate compared to T_b.

T_b [ms]	50	100	150	200	250	300
N slot errors	328	27	20	6	5	5
Success rate [%]	74	97	98	99	99	99

3.6 Comparison between LoRaWAN and Slotted LoRaWAN

This Section presents a comparison between the standard LoRaWAN and the S-LoRaWAN. The same parameters of the radio configurations and the traffic injected are used for both protocols. In this test, the channel traffic and the corresponding throughput are calculated considering only the payload time (T) as traffic generated, the following conditions applied in Eq. 3.12. Moreover, the tolerance interval (T_b) inserted into the slot time is considered into the protocol overhead. The channel throughput in Table 3.4 describes the effective payload transmitted on the channel; this allows a more accurate comparison without considering the protocol overhead. Referring to the traffic analysis, Table 3.4 shows the comparison between the S-LoRaWAN performed with the S-ALOHA MAC (Fig. 3.13 point 1) and the LoRaWAN standard protocol (Fig. 3.13 point 2). Interestingly, the generated traffic resulting from the two protocols is slightly different; this is partly due to the statistical evolution of transmissions; in fact, in the case of the standard LoRaWAN, a frame overlapping is much more probable, with consequent recourse to the backoff algorithm and an average reduction in the number of packets transmitted. In Table 3.4, the measured throughput of S-LoRaWAN and standard LoRaWAN is respectively 15% and 2.6%, finally, with an improvement of 5.8×. This result is higher than the theoretical 2× between S-ALOHA and P-ALOHA; indeed, it has to be considered that the maximum throughput is not placed at the same G (Fig. 3.13), and with the parameters used in this test, the P-ALOHA channel is heavily crowded with a TX success rate of 7%.

Table 3.4: Performance comparison between S-LoRaWAN and LoRaWAN.

	Slotted LoRaWAN	LoRaWAN
Test duration [s]	3524	9596
Packets sent	2894	6817
N of end nodes	24	24
TX period [s]	13	13
TX success rate [%]	33	7
Traffic generated (G)	0.455	0.393
Channel throughput (S)	0.150	0.026

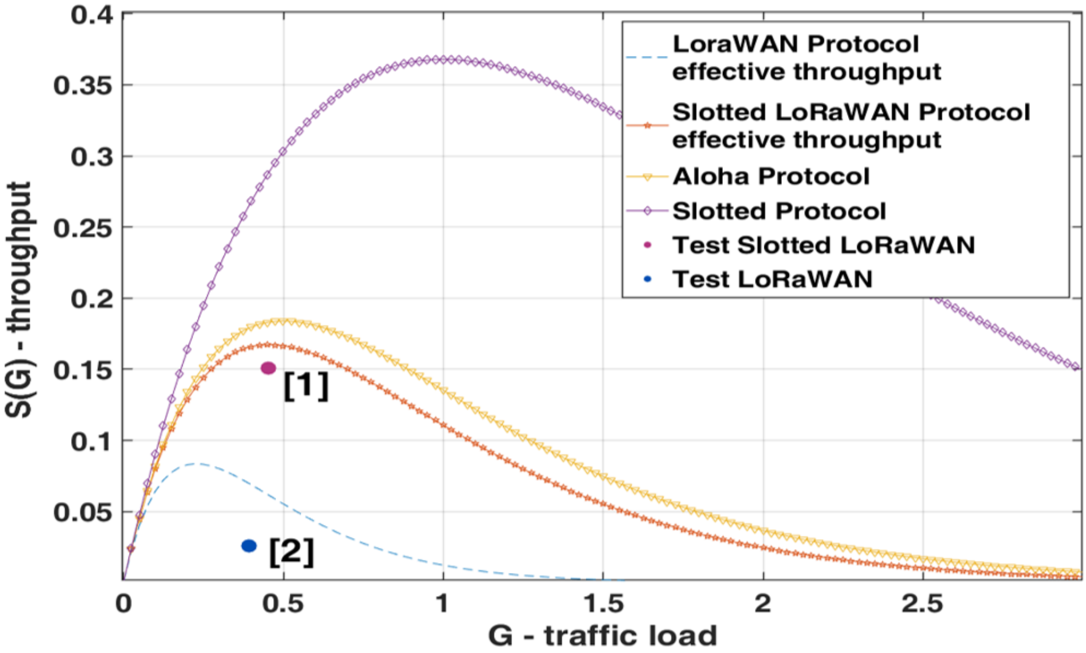


Figure 3.13: Comparison between LoRaWAN and Slotted LoRaWAN. Point [1] refers to S-LoRaWAN and [2] is the standard LoRaWAN throughput.

3.7 NB-IoT: introduction

Following the rapid IoT market expansion, LPWAN has become one of the faster-growing areas in IoT. Low Power Wireless Area Network (LPWAN) is the common term to identify the wireless technologies that enable wide-area communication at low cost and low power consumption. The LPWAN typical application scenario needs to transmit a few bytes with a long-range. Many LPWAN technologies are emerging in licensed and unlicensed markets, such as LoRa, LTE-M, SigFox, and Narrow-Band Internet of Things (NB-IoT). Among them, LoRa and NB-IoT are the two leading technologies [98, 99].

On the cellular networks side, the 3rd Generation Partnership Project (3GPP) has developed the Narrow-Band Internet of Things concept as part of Release 13 [99]. To improve energy efficiency, NB-IoT combines the benefits of the 4G mobile network, namely the global coverage and the long-range, with the energy efficiency typical of LPWANs. Moreover, NB-IoT is intended to provide more reliable indoor coverage for a massive quantity of low-throughput devices [100], serving the high-value IoT market share that pays for high quality of service and very low latency [101, 102]. In contrast, LoRaWAN is targeted to lower-cost devices, with occasional communication needs, very long-range (high coverage), and extended battery lifetime requirements. Today, both NB-IoT and LoRaWAN offer low power consumption and long-range connectivity with the primary aim of being employed as a wireless solution for IoT.

This Section main objective is an experimental evaluation with in-field measurements of NB-IoT vs. LoRaWAN for IoT applications. The comparison between the two protocols is performed in terms of power consumption, energy per bit, battery lifetime, and deployment cost. A wireless sensor network designed for Structural Health Monitoring (SHM) was employed to achieve a realistic comparison. SHM allows evaluating the above mentioned challenges that are considered the primary obstacles for LPWAN deployments [101, 103].

In the long-range communication domain, the most popular protocols are Sigfox, LoRaWAN, and NB-IoT [101, 103, 104]. Sigfox allows remote transfer between devices and an access point through ultra-narrow band modulation, applying stringent uplink and payload size constraints. Sigfox is very similar to LoRaWAN in terms of power consumption and range [105]. However, it is not included in this paper as it is a proprietary protocol, it is less used in IIoT due to its limited payload size (12 B) [104], and for the transmission restriction of 140 B/day and 4 bytes/day for uplink and downlink respectively [103]. The LoRaWAN open standard enables large scale deployments through LoRa, a chirp spread spectrum modulation, with a communication range up to 15 km at low power operation. Many scientific works, such as Section 3.1, describe and model the energy performance for LoRaWAN [76] and the related scalability issues [106].

The NB-IoT [99] is a variant of LTE (4G Long Term Evolution) developed to fulfill the IoT requirements in civil and industrial applications: coverage extension, long battery lifetime, backward compatibility, and user equipment cost reduction are common objectives [107]. The energy performance of NB-IoT is dependent on a multitude of parameters related to the country settings and network operator requirements that can drastically change the end-device average power consumption. In [108] the authors show the NB-IoT independence between the transport block size and power consumption. They vary the payload size between 50 and 100 bits, and the measured power consumption is 716 mW on average. The energy used to join the network is 11.1 J with a connection time of 36 s. In the following experiments, the same independence is confirmed compared with the LoRaWAN protocol. Low power and lifetime are crucial for wireless end-devices and sensor nodes in IoT and other applications [101, 109]. In [101], a LoRaWAN comparison analyzes several factors, such as QoS, latency, network coverage, cost and, scalability, based on the data declared by the developers, but without an actual practical test. They compare both protocols in various use cases, to ensure that LPWAN technologies can provide efficient connectivity solutions across critical and massive IoT deployment, determining their feasibility for specific applications. This study extends and complements this comparison, also providing in-field experimental measurements of the two protocols. Technical differences between NB-IoT and LoRaWAN are summarized in multiple scenarios, such as smart farming [110], manufacturing automation [111], smart building, and logistics [112]. These studies show that both protocols can coexist in the IoT market: LoRaWAN will serve as the low-cost and very long-range deployments, with infrequent transmissions and heavy constraints in battery life. In contrast, applications requiring low latency and high quality of service, in addition to an international coverage [113], will make use of NB-IoT. The outcomes about NB-IoT, in [108] and [101] show 13 years of operability with one transmission (TX) per day and 250 days if a packet is sent every hour in power save mode. These numbers decrease drastically, to 126 and 88 days, respectively, if the extended discontinuous reception is enabled. Finally, [101] concludes that NB-IoT power profile currently leaves open questions on the battery life in real deployments.

The following sections present accurate in-field experimental measurements of LoRaWAN and NB-IoT at the same conditions, allowing a direct in-field comparison. Moreover, this study gives insights on the motivations behind the main similarities and differences, rooted in the architecture of the underlying communication protocols, and it details the key aspects.

3.8 NB-IoT: technology overview

3.8.1 NB-IoT

NB-IoT, also known as LTE Cat-NB1(NB2), is a novel protocol standardized by 3GPP [99]. It belongs to Low Power Wide Area (LPWA) technologies that could work virtually anywhere when infrastructure is present. NB-IoT can operate in three different modes: in-band inside the occupied bandwidth of a wideband LTE, stand-alone as a dedicated carrier, and within the guard-band of an existing carrier [107]. In the second mode, NB-IoT can occupy one GSM channel (200 kHz) while for in-band and guard-band deployment, it will use one Physical Resource Block of LTE (180 kHz). NB-IoT uses the orthogonal FDMA in the downlink and single-carrier FDMA (frequency division multiple access) in the uplink and applies the QPSK (quadrature phase-shift keying modulation) [107]. Each message can reach 1600 bytes of payload. The maximum data transmission rate is limited to 20 kbps for uplink and 200 kbps for downlink. As discussed in [114], NB-IoT is designed for long-life devices and targets a battery life of more than 10 years when transmitting 200 bytes per day. To achieve these performances, NB-IoT uses the LTE energy-saving mechanisms, extending the timers period to minimize energy consumption. Extended Discontinuous Reception (eDRX) and Power Saving Mode (PSM) are the two energy-saving features. In case of devices that rarely uplink and a low need to receive messages, eDRX[5] feature can significantly decrease the power consumption, shown in Figure 3.14. There are two different settings according to the state of the devices: Connected-eDRX or Idle-eDRX. When a device is connected and there is no traffic, it alternates active listening and sleep periods. This behavior is maintained for the Inactivity Timer duration (Fig. 3.14). Otherwise, when a device is idle, the network or the base station cannot request new transmissions. Still, the downlink channel is tracked at Paging Window (PW) events to keep network synchronization and discover if downlink data is pending. The time between two PW is equivalent to an Idle-eDRX cycle (Fig. 3.14).

The PSM is the deep sleep state, shown in Figure 3.14 and defined in 3GPP Rel.12. It allows a reduction of the current consumption, maximizing the amount of time that a device can remain in an extremely low power mode during periods of inactivity. After a wake-up, where data transmission generally occurs, it moves to the idle state, where reception windows allow downlink communication from the base station. The reception phase lasts according to the network policies agreed during the registration process. Following, upon expiry of the timer T3324, the device switches in PSM. In this state, any receiving communication is disabled, but the device remains registered on the network,

---

[5] For detailed information on eDRX see [115].

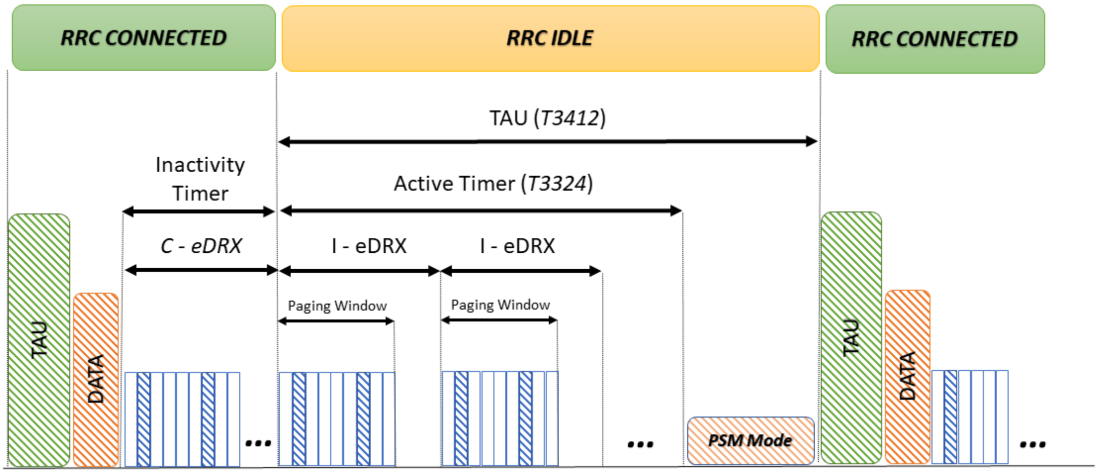


Figure 3.14: Extended Discontinuous Reception: the periodicity value of the reception windows can reach 10.24 s in Connected and 2.91 hours in Idle state, respectively. Power Saving Mode: device remains registered with the network, and it is not necessary to re-attach or re-establish the connection. The maximum duration of the PSM mode is 310 hours.

and re-joining is unnecessary when it switches back to transmit. The timer T3412, set by the device following the network policies, is responsible for managing the PSM mode, enabling the periodic Tracking Area Update (TAU) procedure. The device can disable the PSM at any time if it needs to send a message.

3.8.2 Platform setup

In this Section, a low-power wireless sensor (Section 6.1) developed to measure the cracks in reinforced concrete structures is used as a reference platform to compare NB-IoT and LoRaWAN protocols [26]. This sensor has been designed to guarantee a high sensitivity, up to 5 μm, combined with an extended battery lifetime, which must be at least ten years measuring and sending data ten times per day. A wireless sensor node critical aspects are the radio budget link, power management, and analog front-end. The sensor node embeds an STM32F373 microcontroller, an analog front end, and two radio modules: LoRa and NB-IoT are operated in a mutually exclusive fashion. The MCU handles the analog and digital parts through the integrated Sigma-Delta ADC converter and the serial peripheral interface. A power supply circuit manages a Li-MnO$_2$ lithium battery (4.2 V - 1000 mAh) with 80% efficiency. The SX1276 from Semtech [116] controls the Lora Physical layer and packet buffering. This component achieves a sensitivity of -148 dBm with output power up to 20 dBm, enabling a 168 dB maximum link budget. The NB-IoT transceiver is the SARA-N211 from U-Blox. It is provided in the small LGA form factor (16.0 x 26.0 mm, 96-pin). The module offers data communication over an extended operating temperature with low power consumption, 3 μA in deep-sleep, and 220 mA in transmission at 23 dBm. With a receive sensitivity of -135 dBm, it offers a 158 dBm of link budget. Finally, the M41T82 from ST Microelectronics, an ultra-low-power real-time clock, wakes up the sensor node only at the scheduled time, requiring only 365 nA @ 3 V.

The sensor node draws an average of 23 mA @ 3 V per second in the active mode, used to sample, filter and encrypt the acquired data; requiring an energy of 70 mJ (E_{sensor}). Afterward, the MCU determines which communication protocol will be used depending on the user and application request. Reducing the wireless communication energy is very valuable for low power devices. Indeed, it is well known that the radio transceiver is one of the components with the highest power consumption, as shown in [117]. For each sample, the MCU generates 12 bytes of data, which can be stacked in one buffer or sent immediately to the application server.

3.9 Experimental results

This section presents the experimental evaluation of the SX1276 and SARA-N211 modules in the above-mentioned section 3.8.2. It mainly focuses on the sensor energy performance with multiple payload sizes and coverage conditions to determine the battery lifetime. The sensor node periodically transmits an uplink message, including a single sample or multiple acquisitions queued in one packet.

3.9.1 LoRaWAN end-device analysis

To realistically define the energy profile, a model based on measurements from a real LoRaWAN testbed and previous works [26, 30, 94, 109] was developed. A periodic behavior for each transmission is assumed as a pre-requisite, with a fixed time interval. Therefore the power consumption during one period is studied, including the packet generation, the cryptography, the uplink transmission, the RX1 Delay, and, finally, the downlink window used to receive the acknowledge (ACK). Each datarate used in this evaluation, from 0 to 5, generates several configurations that impact the LoRa modulation. For example, the Equivalent Bit Rates (EBR) of DR0[6] and DR5 are respectively 292 and 5469 bps (Table 3.6); moreover, the transmission time can fluctuate between 225 ms to 4 s with 100 bytes of payload. Such variability impacts the communication range and the power consumption; therefore, smart management of these parameters is crucial to keep the node powered as long as possible. In Table 3.5, the LoRaWAN time

---

[6] It is the equivalent term of DR = 0, SF = 12, or CR = 4

Table 3.5: LoRaWAN Packet time on air

DR	**SF**	**BW** [kHz]	**EBR** [bps]	**T**$_{rx}$ [ms]	**T**$_{tx10}$ [ms]	**T**$_{tx50}$ [ms]	**T**$_{tx100}$ [ms]
0	12	125	293	598.02	1318.91	2629.63	4268.03
2	10	125	977	149.50	370.69	780.29	1312.77
5	7	125	5469	18.69	66.82	148.74	251.14

on-air regarding the payload size (10-50-100 bytes) is presented. Since LoRaWAN needs to transmit the node MAC to identify the packet on the server-side, the transmission time also considers 13 bytes of overhead. The preamble (N$_{pre}$) and Coding Rate (CR) symbols are 8 and 4/5, respectively, plus the CRC (Cyclic Redundancy Check) that is disabled. Finally, the bandwidth is configured at a value of 125 kHz. Under ISO/IEC ISM European guidance, LoRaWAN restricts the packet size with a maximum of 51 bytes for DR0 and DR1, and up to 242 for DR5; moreover, considering the protocol overhead, the payload size is limited to 38 and 229 bytes, respectively with DR0 and DR5. In [94], a study on LoRa SFs assignment is presented. Authors show that underestimating the SF may increase the packet error rate (PER) due to low SNR, and an overestimate can significantly decrease the battery lifetime. Applying a PER strategy, where each sensor node assigns the lowest SF for which the PER falls below a fixed threshold, with a 0.01 PER lower limit [94], the SFs are allocated about 43% SF12, 20% SF11, 12% SF10, 8% SF9, 6% SF8, and 11% SF7[7]. The maximum packet size of 51 bytes for all the configurations is considered in this work since most sensor nodes are in the high SF zone, a value that defines the limit of the maximum number of samples per packet, corresponding to three crack measurements. In [97] and [76], the authors show the correlation between packet loss and network traffic: they indicate a performance drop of around 10% for architectures with 1000 nodes, 36% for 5000, 59% for 10000. Also, [30] shows the effect of saturating the available airtime with one gateway and many nodes. With the proposed SF assignment, the PER increases significantly when the number of devices exceeds 5000. They simulate an upstream scenario with a data period of 6000 s and 21 B of payload. A recent study [30] evaluates the packet loss under challenging environments, such as indoor industrial establishments and a data center facility. In these conditions, the packets received with the wrong CRC vary between 0.5% and 6%. Hence in this SHM testbed, the PER is not negligible and must be taken into account to estimate the average energy consumption.

Table 3.6 presents the measured payload Energy Per Bit (EPB) with different DRs and sizes, considering the power used in TX, in RX and the energy used by the MCU to encrypt and decrypt the data: EPB1 refers to 1 sample (12 B), EPB2 contains 2 (24 B) and EPB3 3 (36 B). The last three columns present the overall Energy Per Packet (EPP)

---

[7] It is the equivalent term of SF = 7, or SF = 6

for a LoRaWAN transmission with different DRs and queue lengths: Packet 1 includes only one crack measurement (12 bytes of payload) whereas Packet 3 is composed of three. Moreover, Table 3.6 shows that the DR0 uses 22× more energy in comparison with DR5. As expected, the EPB does not scale linearly with the payload due to the high ratio between preamble and payload size. For example, with 12 bytes and DR5, the preamble length is 35% of the overall time of air, and with 36 bytes, it is only 24%. This result confirms that buffering the samples in one placket increases the transmission efficiency.

Table 3.6: LoRaWAN EPB & EPP

DR	**SF**	**EBR [bps]**	**EPB 1 [mJ]**	**EBP 2 [mJ]**	**EPB 3 [mJ]**	**Packet 1 [mJ]**	**Packet 2 [mJ]**	**Packet 3 [mJ]**
DR0	12	293	6.69	5.31	4.00	641.28	1017.60	1152.01
DR2	10	977	1.68	1.30	1.01	161.28	249.59	290.88
DR5	7	5469	0.30	0.23	0.16	28.32	43.2	46.08

To carefully model the sensor node behavior, the energy consumption for the first connection and authentication with the LoRaWAN server is measured; this procedure exchanges the cryptography keys and establishes a secure connection between devices. The values measured for DR0, 2 and 5 are respectively 581.29 mJ, 172.25 mJ, and 62.03 mJ. EPP in Table 3.6, and the equivalent T_{Packet} in Eq. 3.15, take into account the uplink packet (T_{tx} - Eq. 3.16) formed by the payload (PL), preamble and 13 bytes of LoRaWAN overhead, the waiting period (T_{rx1}) between the uplink and downlink windows and lastly, the receive period used to detect the ACK (T_{rxw}).

$$T_{Packet} = T_{tx} + T_{rx1} + T_{rxw}, \tag{3.15}$$

$$T_{tx} = \frac{2^{SF}}{BW} \cdot (N_{pre} + 4.25 + N_{PHY}). \tag{3.16}$$

T_{tx} expresses the time in seconds required to transmit both the preamble and the payload; the latter is composed of the number of symbols calculated in Eq. 3.6.

3.9.2 NB-IoT end-device analysis

This section focuses on the NB-IoT energy performance of the sensor node in the same deployment conditions as the previous subsection. As it is not trivial to estimate the energy consumption of the transmission due to the multitude of NB-IoT parameters, such as the eDRX and PSM timers, the transmission power and the number of repetitions requested by the network, a model based on measurements from a real NB-IoT testbed, and previous works [101, 108] is combined to precisely derive the NB-IoT energy profile. The SHM sensor node is tested by varying the payload and the RSSI that influences the module power consumption. Precisely, we define the -80 dBm average RSSI as Good (G), -110 dBm average RSSI as Medium (M) and finally, -130 dBm average RSSI as Bad (B). Table 3.7 shows the measurements of energy per packet and T_{active} with 10, 50, 100 and 400 bytes of payload, depending on the 3 defined coverage levels. Column E_{mean} results from 50 successive measurements with the same RSSI condition to model the average energy performance for each of the 12 presented tests. To estimate the battery lifetime, the RSSI could be considered one of the most relevant factors. Indeed, the absence of correlations between energy and payload size (Table 3.7 - N bytes) can be appreciated dividing the values in Table 3.7 for coverage conditions. Indeed, among *(a)* and *(d)* the T_{active} and E_{mean} differences are 2% and 10% sending 40× more bytes. Similar behaviour can be detected in B coverage, between tests *(i)* and *(n)*, where the T_{active} ranges between 37.2 s in *(m)* and 46.6 s in *(i)*; the E_{mean} is included in a 25% of variability.

These measurements have been carried out with the Swisscom network provider, which releases the default 3 minutes period for T3324, whereas the T3412 can be set up to 310 hours, avoiding TAU signaling between successive uplinks. The T3324 energy consumption must be added for each transmission because the SARA-N211 module is awake in listening mode. The overall value for 3 minutes timer is 844 mJ, equal for each coverage condition. The maximum energy measured in G condition (test *(a)*) is 6× higher compared to minimum, and the *(n)* test maximum energy is 37× the test *(b)*. Analyzing Table 3.7 and Fig. 3.15, a significant increase of the variance in B than M and G coverage is detected. These results reveal the high power consumption variability, which is not under direct user control. Indeed, each network provider manages differently the network parameters, such as the number of repetitions, the transmission power, TAU, and eDRX timers. For future designs, Table 3.7 - I_{max} is a useful tool for power management calculations. The good coverage group, in green, has an average RSSI

Table 3.7: NB-IoT Energy Characterization

ID	**C**	**N bytes**	**$T_{act.}$ [s]**	**I_{max} [mA]**	**E_{mean} [mJ]**	**E_{max} [mJ]**	**E_{min} [mJ]**	**RSSI [dBm]**
a	G	10	11.9	138	2063	3007	517	-83
b	G	50	11.9	146	1858	3111	486	-81
c	G	100	12.0	135	1856	3240	499	-75
d	G	400	12.2	138	2067	3232	550	-75
e	M	10	13.7	245	2677	4549	1847	-112
f	M	50	12.8	232	2453	4078	1890	-109
g	M	100	12.6	219	2379	4150	1903	-110
h	M	400	12.8	225	2386	3786	1972	-107
i	B	10	46.6	151	9047	17072	5453	-130
l	B	50	41.1	175	7641	16298	5579	-136
m	B	100	37.2	169	6818	13264	5200	-135
n	B	400	40.5	185	7552	17845	5745	-134

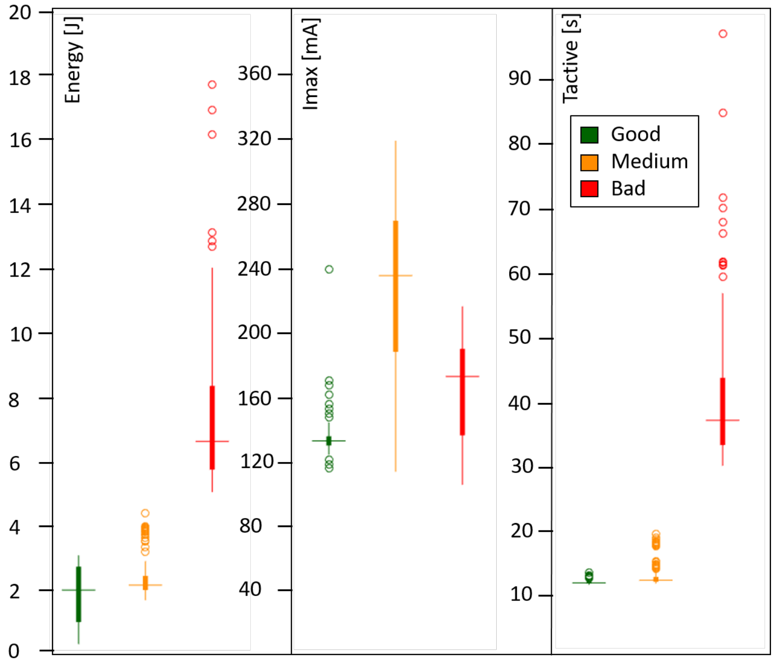


Figure 3.15: NB-IoT characterization with median, 25th and 75th percentiles. Good (G) in green with an average RSSI of -80dBm, Medium (M) in orange with an average RSSI of -110dBm and, Red (R) with an average RSSI of -130 dBm.

of -80 dBm; this generates a mean T$_{active}$ of 12 s; with these parameters, the average energy for each packet is 1982 mJ. In the M group, the T$_{active}$ slightly increases, with a mean of 13 s, but the resulting energy 2474 mJ grows of about 25% in comparison with good coverage; indeed, the maximum current is 100 mA higher. This behavior means that the NB-IoT cell increases the output power before raising the number of retransmissions. In analogy with LoRa, the NB-IoT's T$_{active}$ is highly correlated with the communication latency that for the latter reaches up to 46 s in worst cases (Table 3.7). Tests (i),(l),(m),(n) are close to the maximum sensitivity of the module, the resulting energy, and T$_{active}$ grow heavily: the average time is 41 s with a maximum of 17845 mJ and, a medium of 7765 mJ.

Figure 3.15 presents the statistical analysis of the energy, I$_{max}$ and T$_{active}$ features showing the median, 25th, and 75th percentiles of all the data acquired (600 samples). The energy grows with respect to the received RSSI decrease, which results from the T$_{active}$ and I$_{max}$ combination depending on the coverage strength and the network request. Indeed, the NB-IoT protocol raises TX output power before increasing the number of retransmissions and the correlated T$_{active}$. The packet time difference between G and M is negligible, but B T$_{active}$ is at least 3× compared to G. Furthermore, in M the output power correlated with the I$_{max}$ is 2× and 1.3× compared with G and B respectively, but the T$_{active}$ is still comparable with B.

Table 3.8: NB-IoT EPB

C	**EPB 1 [mJ]**	**EPB 2 [mJ]**	**EPB 3 [mJ]**	**EPB 8 [mJ]**	**EPB 33 [mJ]**
G	29.4	14.8	9.8	3.6	0.9
M	34.5	17.2	11.5	4.2	1.0
B	89.6	44.9	29.9	11.2	2.7

The energy used for the first connection and authentication with the NB-IoT cell is checked; this procedure subscribes the sensor node on the network. The values measured for G, M, and B are respectively 15843, 17182, and 19124 mJ with an average connection time of 80 s. NB-IoT enables a packet length up to 1600 bytes [118], but the used module (with firmware version: 0.6.57, A07) is limited. Consequently, the queue is restricted to 33 samples, each consisting of 12 bytes. In Table 3.8, the payload EPB with different coverages and sizes is presented: EPB 1: 12 bytes of payload (1 sensor sample); EPB 2: 24 bytes of payload (2 sensor samples); EPB 3: 36 bytes of payload (3 sensor samples); EPB 8: 96 bytes of payload (8 sensor samples); EPB 33: 396 bytes of payload (33 sensor samples). The EPB in Table 3.8 includes the uplink energy used in T3324 and T_{active}: it suggests that the EPB decreases increasing the queue size, as also confirmed in the recent work [108].

Compared to LoRaWAN, sending one sample per packet with NB-IoT reduces battery life, as presented in the following subsection. Moreover, results suggest that T_{active} is strictly associated with the coverage condition and does not depend on payload length, i.e., the average RSSI. The NB-IoT protocol increases the number of retransmissions from 32 to 2048 when the RSSI is low, a setting not managed by the end device. Moreover, the power consumption is independent from the uplink and downlink data rate, which is mainly affected by the number of rentransmissions [108]. In static working conditions, the energy consumption among packets varies with respect to network parameters: the output power, the number of retransmissions, and the T_{active} can be modified between successive uplinks and are not under the direct control of the module. To prove the NB-IoT inefficiency for sporadic and tiny transfers, Fig. 3.16 presents differences between single and multiple packets in a single connection. Taking as a reference the Test (*d*), the T_{active} and the E_{mean} are evaluated sending one (Pkt1) to ten (Pkt10) successive packets with 400 bytes of payload in G coverage.

In contrast to LoRaWAN, the energy does not grow linearly with the number of uplinks in a single connection (Figure 3.14 - DATA), but it only increases by 11%, sending 10 times more bytes. In the Pkt10 condition, the EPB is about 0.1 mJ, 9x less than the EPB 33 presented in Table 3.8. However, a buffer of 330 samples could generate an excessive latency for many applications; hence, in this study, the EPB is compared

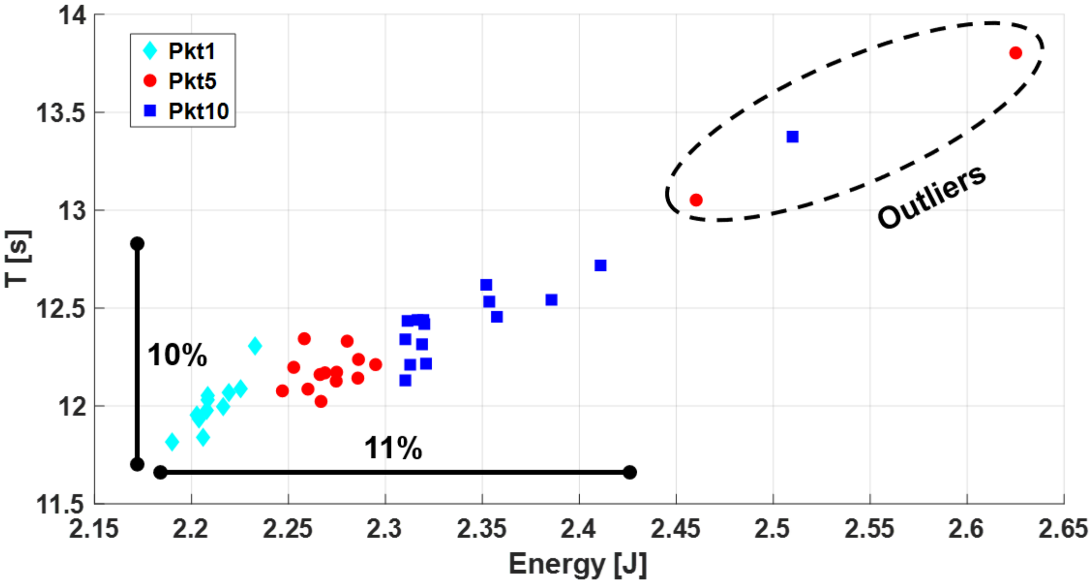


Figure 3.16: Differences between single and multiple packet transmissions in a single connection.

considering only one transfer for each connection and commonly used in a deployment where sporadic transmissions are required.

$$E^*_{mean}(C) = E_{mean}(C) \cdot (1 + N_{pkt} \cdot 0.01). \tag{3.17}$$

Finally, the expected E^*_{mean} generated in a single connection where multiple packets are transferred, is presented in Eq. 3.17. The C variable points at the coverage condition energy in Table 3.7 - E_{mean} and N_{pkt} is the number of packets transmitted together.

3.9.3 Battery life and comparisons

This subsection focuses on estimating the battery life in the SHM application scenario based on the above-presented power measurements. One of the most challenging features is to achieve a lifetime of 10 years. In this evaluation, pre-requisites impose that each node is equipped with a 1000 mAh lithium battery @ 3 V, a widely used type of battery for SHM nodes [26]. Thus, each sensor sampling energy consumption is constrained, and its usage is regulated by the energy per packet and the queue length. For the estimation, it is considered the energy used for the initial connections ($E_{connection}$) calculated in previous sections with 10 samples per day (N_{tx}) to fulfill the plots in Figure 3.17. In particular, based on previous considerations, the average packet loss changes considerably depending on every single deployment, varying between 0% to 60% due to crowded radio channels or noisy electromagnetic environments. Hence it is misleading to provide a single result for each configuration. The packet loss probability for energy estimation in the LoraWAN case study is included and the Figure 3.17 takes into consideration the effective communication variability, providing a lower and upper bound between 0-60% ($P_{PktLoss}$). A conservative parameter considers the integer bar to estimate the average sensor lifetime span depending on the queue and DR configurations. Eq. 3.18 and Eq. 3.19 show the formulas used to calculate the data in Figure 3.17 for LoRaWAN and NB-IoT. T_{LoRa} and T_{NB-IoT} provide the times in days, E_{SLEEP} is the sleep energy calculated with a 365 nA current. Lastly, C and Q select the coverage and queue configurations from Table 3.6 and Table 3.8.

$$\begin{cases} T_{LoRa} = \frac{(DCDC_{eff} \cdot E_{batt}) - E_{connection}(C)}{[E_{LoRa} + E_{sensor}] \cdot \frac{N_{tx}}{Q} + E_{SLEEP}} \cdot 86400\ s \\ E_{LoRa} = \left(12 \cdot Q \cdot 8 \cdot EPB(C,Q) \cdot \frac{1}{1 - P_{PktLoss}}\right) \end{cases} \tag{3.18}$$

$$T_{NB-IoT} = \frac{(DCDC_{eff} \cdot E_{batt}) - E_{connection}(C)}{[E^{*}_{mean}(C) + E_{sensor}] \cdot \frac{N_{tx}}{Q} + E_{SLEEP}} \cdot 86400\ s. \tag{3.19}$$

The resulting lifetime is less than 10 years with Packets 1-8 for both protocols in DR0/Bad coverage (Figure 3.17), but it is interesting to notice that with Packets 1-3 LoRaWAN reaches this threshold in DR2 and DR5. NB-IoT allows this duration only with Packet 33, in all coverage conditions; on the other side, LoRaWAN reaches the target from DR2 without queuing. If the application requires a transmission for each sample, the expected lifetime is respectively 4.5 months and 3.5 years for NB-IoT and LoRaWAN in the worst case. As shown in Figure 3.17, with equal coverage, NB-IoT EPB is one order of magnitude higher than that measured with LoRaWAN. The LoRaWAN EPB decreases more if coverage improves compared to buffering techniques, as opposed to NB-IoT, where the decrease is similar. Finally, the only cases where EPB is advantageous for NB-IoT is when the coverage is at least DR2/M and the message sent contains 33 samples (Packet 33).

3.9.4 Quality of service, cost and coverage

Wireless communication energy consumption is one of the principal issues in IoT applications. Nevertheless, many factors should be considered, including the QoS, the cost, and the coverage. This section highlights that LoRaWAN works on unlicensed ISM channels with an asynchronous protocol. In crowded channels and industrial environments, packet loss cannot be considered a negligible factor, given that it can decrease the expected battery lifetime up to 37%. On the other hand, NB-IoT offers an optimal QoS, with reliable data delivery, working on licensed spectrum, and an LTE-based synchronous protocol. However, its communication latency is not optimal. Indeed, the maximum LoRaWAN packet time, which corresponds to the transfer delay, is 2630 ms with DR0. It is ≈ 17× lower than the NB-IoT's T_{active} in B coverage (Table 3.7 - *i*). Different parameters must be examined for the implementation costs. A generic NB-IoT

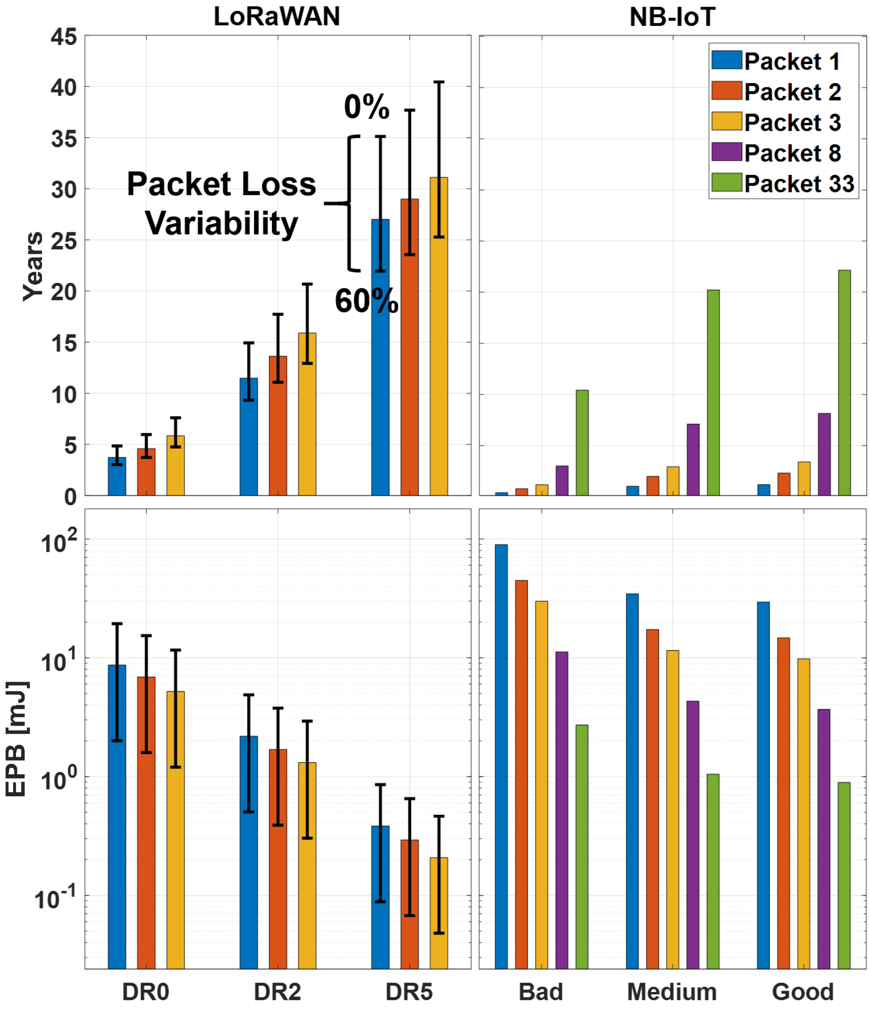


Figure 3.17: Expected battery lifetime and EPB with LoRaWAN and NB-IoT. End-device coverage is divided into: DR0/Bad with an average RSSI of -130 dBm, DR2/Medium with an average RSSI of -110 dBm and DR5/Good with an average RSSI of -80 dBm.

module can exceed 20€ compared to 3-5€ of a LoRa transceiver [101]. Moreover, it is important to consider the cost related to traffic generated by each device (500 MB of traffic are priced today at 10€). This amount of data are more than enough for the entire sensor life in a typical monitoring application. On the other hand, a LoRaWAN network must have at least one access point (300€/gateway) and the server (1000€/base station). In the SHM application, the system generates 120 bytes daily, allowing more than 100 years of hypothetical operation with a single subscription. In summary, Eq. 3.20 quantifies the deployment cost of the two technologies.

$$\begin{cases} Cost_{NB-IoT} = (Cost_{module} + Cost_{SIM}) \cdot N \\ Cost_{LoRa} = Cost_{module} \cdot N + Cost_{Gateway} + Cost_{Server} \end{cases} \tag{3.20}$$

Whenever the number (N) of sensor nodes is a few tens, the NB-IoT is more affordable due to the high installation cost of LoRaWAN gateway and server, as shown by Eq. 3.20, enabling quicker time to market (TTM) in regions where the LoRaWAN is not deployed yet. On the other hand, LoRaWAN is today more affordable for large-scale deployments due to NB-IoT module higher cost. When the TTM is a concern, NB-IoT has an advantage because of network operators' plug-and-play service. Moreover, for national scale coverage applications, for example, in the monitoring of transportable goods to determine the pallet locations on highways or railroads, NB-IoT will be the only solution due to the infrastructure already provided by the network operators. To cover limited or remote areas where network operators do not offer adequate coverage, LoRaWAN devices with dedicated support can instead be more efficient.

3.10 A multi-protocol system for configurable data streaming: introduction

Wearable and portable healthcare devices are among the most rapidly growing in the IoT marketplace and have gained momentum in digital health and biomedical applications [119]. Monitoring, tracking, and classifying the vital human signs is a key megatrend for consumer electronics to improve the health condition and wellness of users and their families [120]. Portable healthcare devices are today battery powered and the energy autonomy, together with a small and comfortable form factor, are the most critical design constraints as they can be worn on the body, arms, and legs/feet.

Synergistic hardware-software design approaches are needed when designing these highly constrained devices, which are usually focused at the node level, with smart power management techniques. These approaches are based on hardware and feature management, such as the use of low-power radios, custom ultra-low-power components, or reducing the energy consumption of the primary and most frequent tasks.

One of the most power-consuming tasks is to transfer the data acquired by the node either to cloud servers or to personal gateways (e.g., a mobile phone) for low-latency feedback [121]. Therefore, wireless communication is an essential feature. It is often the bottleneck both for the data connectivity and for the entire system power budget, considering that applications might need to report continuous data. At the same time, users usually move worldwide, indoor and outdoor.

This Section tackles the joint challenge of communication energy minimization and maximization of the communication flexibility under several different connectivity scenarios. Hence a multi-protocol communication system is designed for IoT devices to span the trade-off between datarate and power consumption. The following Section proposes the implementation of a single board with the capability to switch seamlessly between WiFi, Bluetooth, and LoRaWAN. Simultaneously, the IoT application can dynamically select the wireless standard that consumes less energy for the amount of data to transmit, even when constrained by latency requirements.

Furthermore, with a special focus on wide-area connectivity, the newest low-power wide-area network protocols, such as 5G NB-IoT and LoRaWAN, are designed for multi-kilometers coverage area; this feature might be not enough for users living in urban or rural areas. Thus, a multi-hop uplink range extension is designed on the top of the LoRaWAN standard that permits to extend the coverage range to tens of km. Lastly, the power budget of the multi-protocol solution is extensively assessed, characterizing the best communication trade-off in several scenarios, typical of wearable and healthcare applications (i.e., bursty, continuous streaming, sporadic alarms, or periodic reporting without latency constraints) in terms of data amount and distance from the application gateway.

In recent years, wireless monitoring medical devices have been used in the daily life of more and more people. Some of these devices are in the form of accessories such as armbands, glasses, and smart watches. The advancements from earliest WSNs in healthcare [122] have made the health signal monitoring a reality. Indeed, many wireless body-area networks (WBANS) have been proposed that cover real-time and short-range medical information collected from different sensors with low power consumption [123], and wake-up solutions for ultra-low power and very long battery lifetime devices [121]. Several commercial products [124] highlight that Bluetooth (BT) connection is the most commonly used for short-range and low data rate transfers. In this field, BT is used to exchange raw data between sensor nodes and a central node unit (personal gateway) that stores and processes the information, becoming the industrial standard for WBAN bio-sensors. In the IoT for industrial markets, the focus has been primarily on long-range connectivity. The most promising protocol stacks in this area are SigFox, LoRaWAN, and the emerging standard NB-IoT. Even though their noise robustness, the key limitations of long-range WSN are the power consumption and the low data rate. Hence they are used to transmit only essential information. On the other hand, even if wireless personal area networks support data streaming, only high throughput protocols, such as WiFi, are useful to stream high-bandwidth data at lower energy per bit at short range.

Data obtained by sensors is not useful unless something can be done with it; therefore, the collected measurements must be forwarded to a database where relevant parties, such as automated services or human operators, can extract features.

Few commercial products support multi-protocol operation, and most of them are in the IEEE 802.15.4 and 802.11b/g/n field, supporting mainly WiFi and Bluetooth 2.0. The CC1352 [125], from Texas Instruments (TI), is a multi-band wireless MCU. It enables sub-GHz modulations, such as FSK and 2-GFSK, and 802.15.4 protocols at 2.4GHz. Whereas the CC1352 supports dual-band operations, the CC2652 [126] can run multiple protocols in the same band; for example, it supports the coexistence of WiFi and BLE 4.0. However, they are low-power and low-cost devices with limited features and performance, with a maximum bit rate of 2 Mbps, often not enough to support multi-channel bio-medical sensors as [127]. The WL1835MOD [128], from Texas Instruments (TI), offers high throughput transmission along with WiFi and Bluetooth coexistence. It is a 2.4-GHz module with a single multiplexed antenna solution. Unfortunately, TI provides drivers only for high-level operating systems such as Linux and Android. Hence this product is not usable for low power and compact embedded devices with bare-metal firmware. Moreover, all the previous devices do not support long-range communication and cannot enable continuous monitoring in an outdoor environment.

Closest to the proposed approach, the LoPy4 [129] is a quadruple bearer MicroPython board (LoRa, Sigfox, WiFi, and Bluetooth) IoT applications. It embeds an Xtensa dual-core 32–bit LX6 microprocessor, with up to 600 DMIPS, which manages all the stacks and the host interface. The LoPy4 can run all the protocols in parallel without any issues and limitations; the main differences with the proposed multi-protocol system come from the throughput and the power consumption. Indeed, the measured LoPy4's WiFi EPB (Energy Per Bit) is 80 nJ 2× higher compared to Multiradio (i.e., multi-protocol system), with a maximum bitrate of 9.1 Mbps (with UDP). The BLE and LoRaWAN (DR0) need an EPB equal to 3.7 μJ and 24.3 mJ, 5 and 2.7× worse than Multiradio solution.

There are few open challenges in remote health monitoring [119]. Some of these are the security risk related to transporting large amounts of sensitive data, the sensors reliability to ensure that they are monitoring accurately, and the possible disconnection from the central server if the patient runs out the maximum communication range or if the battery is discharged. Moreover, it features an extended version of LoRaWAN, which supports multi-hop communications. Indeed, cities and mountain areas may have dark spots where the sensor cannot reach the healthcare server. The extended LoRaWAN solution outperforms other methods such as [130], which is a LoRaWAN compliant multi-hop uplink extension. In fact, it implements a lightweight multi-hop version with the capability of both downlink and uplink transmission, not supported in [130].

3.11 Extended LoRaWAN (E-LoRaWAN)

The system aims to allow the out-of-range node to reach a specific healthcare server through a multi-hop connection, exploiting a LoRaWAN node already connected to the network as a LoRa repeater, located in the middle between the gateway and the out-of-range node. The LoRaWAN coverage reaches more than kilometers in some cases; nevertheless, it could not be enough for some applications where end-devices move to areas too far away from the nearest gateway, becoming totally disconnected from the network. The LoRaWAN standard does not offer any solution in these conditions, forcing it to deploy a new gateway. To overcome this limitation, a dual-hop range extension protocol is developed as an add-on to increase the network coverage with a cheap and straightforward solution. This work suggests a specific communication system to achieve such a goal. As described in Fig. 3.18, the mechanism starts when the out-of-range node performs the *Join Request* for the first time; concurrently the associated timer starts.

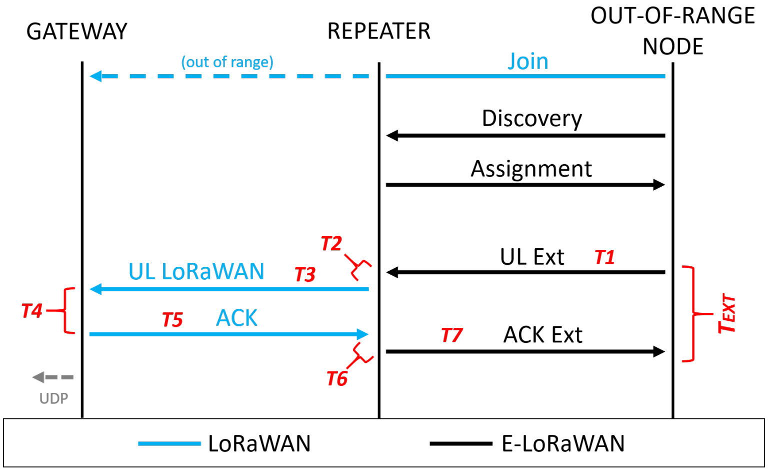


Figure 3.18: Extended LoRaWAN (E-LoRaWAN) timing diagram with multi-hop

After the timeout, if nothing is received, it is assumed the failure of the connection attempt. Then, the node transmits a specific multi-hop join request, named *Discovery*, for the nearest LoRa repeater, which will answer by sending another packet, called *Assignment*, communicating the success of the assignment process. At the end of this part, the repeater will open a new reception window. It can receive packets from out-range-nodes and, eventually, forward their data to the gateway to reach the desired destination. To achieve this operation, the repeater must switch the protocol, taking 24 milliseconds ($T2$) using the standard LoRaWAN, which affects $T3$, $T4$, and $T5$ intervals. Furthermore, the acknowledgment mechanism is implemented to guarantee a specific level of reliability. Therefore, after acknowledgment reception, the repeater changes protocol again to inform the node about this operation success, taking 20 milliseconds ($T6$). At the out-of-range node side, the window for acknowledgment reception is opened after data transmission. This period is equal to 6 seconds (T_{EXT} in Fig. 3.18) using the same parameters for both operations, i.e., DR0, LoRa modulation, SF12 and BW equal to 125 kHz, which influence $T1$ and $T7$.

The maximum number of nodes that a single repeater can support is highly dependent on the application and the single repeater radio configuration. Indeed, due to ISO/IEC ISM regulations, wireless devices working on ALOHA MAC cannot occupy more than 1% of the channel time. Since the time of air depends on the DR, the maximum number of daily packets changes significantly. Assuming a payload length of six bytes, as tested in our deployment, the maximum number of transmissions is 1043, 2880, and 27871, respectively for DR0, DR2, and DR5. But with 50 bytes of payload, the maximum number decreases to 432, 1440, and 8640. As in-depth studied in [24, 25], the packet collision rate (PCR) can reach 60% decreasing drastically the maximum number of transmissions and the number of devices supported by each repeater. T_{EXT} varies between 3 to 6 seconds, depending on the repeater and out-of-range node DR selection. It can be easily calculated taking into consideration the timing diagram in Fig. 3.18. To find the maximum number of supported devices per repeater the developer needs to maximize the N (number of out-of-range devices) for given Eq. 3.21

$$\begin{cases} ((N+1) \cdot N_{TX}) \cdot \frac{1}{PCR} < Max_{TX} \\ \\ ((N+1) \cdot T_{EXT}) \cdot \frac{1}{PCR} < \frac{DAYTIME}{N_{TX}} \end{cases} \tag{3.21}$$

where N$_{TX}$ is the number of UL per day required by the application, Max$_{TX}$ is the maximum number of transmission per day depending on DR selection, as expressed above. Finally, $DAYTIME$ is the number of seconds per day. For example, in the worst case with DR0 and 50 B of payload, the T$_{EXT}$ is 6 seconds. If we suppose an average N$_{TX}$ of 24, the maximum nodes supported by each repeater is 9. Whereas with DR5, the T$_{EXT}$ is 3.6 seconds and the number of nodes increases to 215.

To measure the average Packet Reception Rate (PRR) of each node and to validate the Eq. 3.21, two tests have been made using one E-LoRaWAN repeater for both of them. With DR0 and a scheduled transmission every 20 seconds the repeater supports only two out-of-range nodes, and the ISO/IEC ISM regulation is quickly overtaken. Results show that by increasing the total number of devices to be served, the average PRR decreases. In the first experiment, the PRR is equal to 93% with two nodes, while with four nodes, it is 44.9%.

3.12 Multi-protocol transceiver

3.12.1 MultiRadio

MultiRadio is a multi-protocol wireless IoT transceiver developed at the University of Bologna in 2019. It supports high-speed WiFi 802.11b/g/n, Bluetooth 4.1 Low Energy and the LoRaWAN end node, enabling high throughput and long-range communications. The on-module MCU manages all the stacks, providing an easy-to-use wireless interface as well as fast deployment. The module integrates the STM32F411 MCU from STMicroelectronics, the Cypress CYW4343W, the Semtech SX1276, the power management ICs, and a chip antenna resulting in a small form factor design. It could be interoperable with various vendors' 802.11b/g/n access points in a wireless LAN with seamless roaming capabilities and advanced security. The module provides SPI ports to interface with a host controller. Fig. 3.19 shows the block diagram, whereas Fig. 3.20 presents a picture of the MultiRadio shield for NUCLEO boards. Both hardware and software

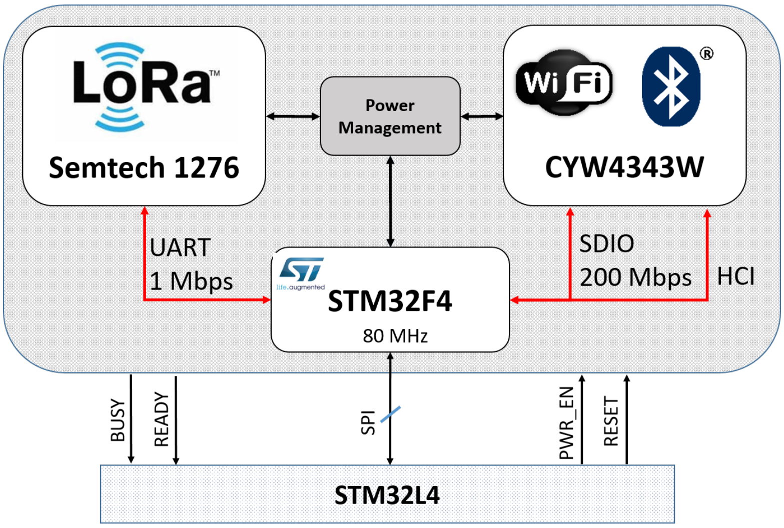


Figure 3.19: MultiRadio transceiver block diagram

projects are open source and available on GitHub[8] [9]. All the conclusions and values reported in this Section are the results of experimental measurements. The MultiRadio's firmware runs on Cypress WICED (Wireless Internet Connectivity for Embedded Devices) [131], an embedded platform for ARM MCU allowing complex IoT applications. WICED is a comprehensive environment supporting WiFi, Bluetooth Classic (BT), and Bluetooth Low Energy (BLE) through the ThreadX RTOS. In a parallel task, the MCU runs the LoRaWAN stack for the SX1276 transceiver, enabling long-range communications. This open-source library [132] supports LoRaWAN 1.0.x, 1.1.x, and 1.2.0 versions, and it has been modified for our E-LoRaWAN with multi-hop transmissions. The Cypress CYW4343W is a highly integrated SoC that needs few external components. It is engineered for smart-phones, wearable devices, tablets, and a wide range of low-power portable devices. The chip includes a WLAN IEEE 802.11b/g/n radio, Bluetooth 4.1 support, and an FM receiver. Also, it integrates a power amplifier, a low-noise amplifier, and an internal transmit/receive RF switch, further reducing printed circuit board area and the overall solution cost. The CYW4343W host interface supports SDIO digital bus, providing a data transfer rate of up to 200 Mbps. Moreover, an independent HCI UART is provided for the Bluetooth host interface to reduce the power consumption when the WiFi is powered off. The WiFi interface has an RX sensitivity up to -92 dBm and maximum output power of 22 dBm. In this condition, the power consumption is between 0.1-1 W depending on bitrate and transmission power. On the other side, the Bluetooth interface has an RX sensitivity of -90 dBm. The average power consumption in slave mode is 95 mW, whereas in low energy mode, it decreases to 844 μW with a beacon every 10 s. Finally, the SX1276 features the LoRa long range modem that

---

[8] https://github.com/GreenWaves-Technologies/GAPuino_BSP/tree/master/multi_radio
[9] https://github.com/tommasopolonelli/MultiRadio

Figure 3.20: MultiRadio shield

provides ultra-long range communication and high interference immunity through the patented spread spectrum modulation developed by Semtech. Due to the high sensitivity of -148 dBm and the RF output of 20 dBm, this transceiver enables a communication range over 10 km in line-of-sight.

The MultiRadio module provides a high-speed (48 MHz) host interface through SPI bus and four control GPIOs. Digital communication is full-duplex; this means that the MultiRadio can receive and transmit in parallel over three wireless protocols. The $BUSY$ pin signals that the module is not ready to receive further commands; the $READY$ pin indicates the presence of a newly received packet and can be used as a wake-up signal for the host, in case of sporadic transmissions. Lastly, PWR_EN and $RESET$ manage the deep-sleep and the reset.

3.12.2 Power consumption profiling

MultiRadio covers a wide area of industrial, biomedical, and consumer applications. Every deployment has different constraints and requirements due to the available energy, the number of bytes generated, the communication range, and the product cost. Because there is no single optimal solution, the wireless protocol can be selected depending on the scenario. Therefore, the MultiRadio can adjust the wireless protocol, but can even help identify the most suitable implementation during prototyping.

In deep-sleep mode, the MultiRadio consumes just 1.65 μW, enabling low power operation with battery-powered devices. The WiFi stack provides two working modes: high-throughput (WHT) and low-power (WLP). The WHT enables a maximum bitrate of 30 Mbps with an average streaming speed of 22 Mbps. That is plenty for high-sampling rate signals, such as images or biomedical multi-channel probes. The WHT

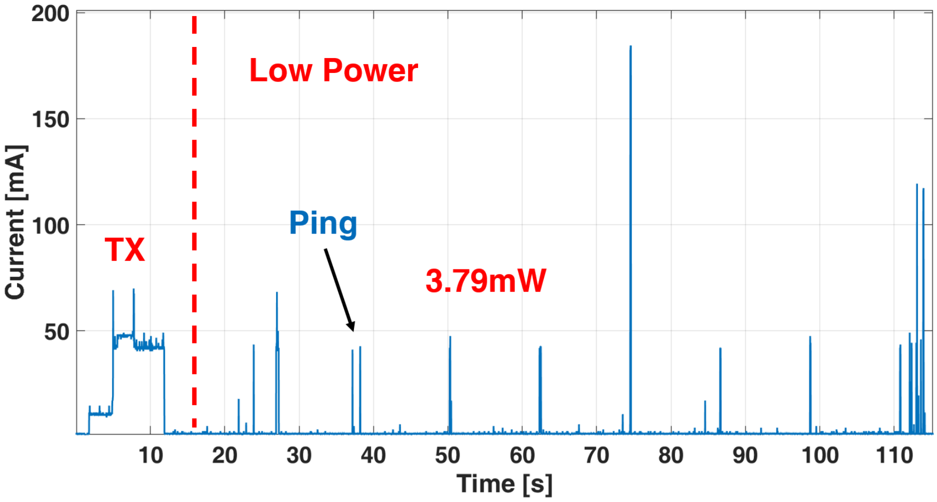


Figure 3.21: Power consumption profile in WLP mode. After a single UL, the MultiRadio keeps the link with the router in low-power mode. Sporadic pings check the connection status.

supports full-duplex data transport and keeps the MultiRadio always connected with the WLAN router. This modality draws 1 W from the power source. When the downlink bandwidth is not a stringent requirement, the WLP can drastically reduce the energy used. However, the bitrate is lowered to 9.1 Mbps in UDP and 4.8 Mbps in TCP/IP. After an initial connection that requires 104 mJ, the average power consumption is 3.79 mW. Indeed, in this mode, the CYW4343W enables the radio only on transmission, and in the remaining time, it is in sleep. With this mode, reception is feasible only after an uplink, and periodic pings must be sent to the WLAN router to keep the connection open. Fig. 3.21 shows the WLP power profile after a single transmission.

The BT stack supports both Bluetooth 2.0 and BLE 4.1. When the device is disassociated, the measured average power consumption is 363 μW, otherwise, it can reach 29 mW and 79 mW, respectively with BLE and BT 2.0. With these configurations, the maximum bitrate is 40 kbps (BLE) and 200 kbps (BT). BLE is used to send data over short distances, with a maximum range of 30 m, 20 m less than BT. This wireless technology has become an alternative for WiFi, especially in IoT devices. Indeed, BLE is more suitable for transmitting small amounts of data, like sensor readings of temperature, alarms, commands, etc. However, BLE is not optimal for sending continuous low-latency and high bandwidth data streams to a server (such as those needed for neural monitoring and feedback applications [127]); thus, if low-latency high-bandwidth data streaming is required, WiFi mode should be used. For example, the beacon energy is lower than WiFi (in WLP mode), sending a 100 bytes packet 6 times per second, an equivalent data rate of 4800 bps. Increasing the payload size to 1 kB decreases the number of packets per second that the BLE can send to be energetically convenient,

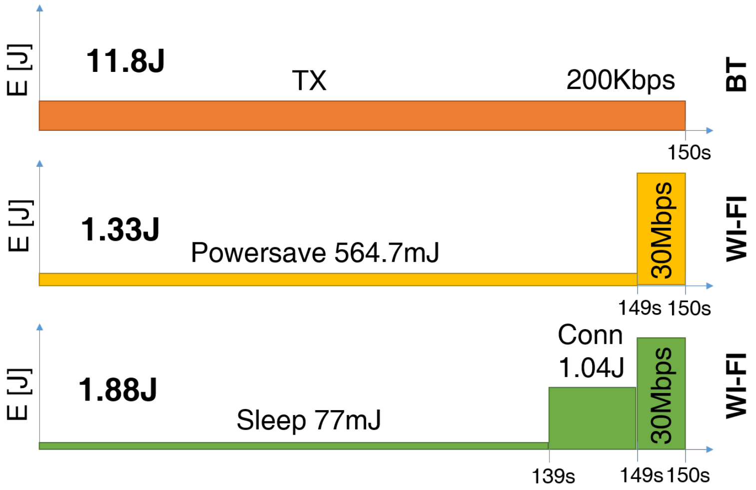


Figure 3.22: Energy consumption comparison between BT in RFCOMM and WiFi, transferring the same amount of data

Table 3.9: LoRaWAN and E-LoRaWAN energy per packet and EBR

DR	SF	BW [kHz]	EBR [bps]	E_{TX50} [mJ]
EXT	12	125	147	2300
DR0	12	125	293	1152
DR2	10	125	977	290
DR5	7	125	5469	46

Table 3.10: Energy per bit (EPB) and equivalent bit rate (EBR)

	WiFi (UDP)	BT (RFCOM)	LoRa DR5	LoRa DR2	LoRa DR0	LoRa EXT
EBR	29.6Mbps	200kbps	5.46kbps	977bps	293bps	147bps
EPB	30nJ	730nJ	280μJ	2.22mJ	8.87mJ	16.9mJ

which is 0.6 packets/seconds. Over this average bitrate, the WLP is more appropriate.

We tested our MultiRadio transmitting the same amount of data with BT in the RFCOMM profile and WiFi to verify the energy and the latency. The result is presented in Fig. 3.22. In 150 s of streaming, the BT's energy is 11.8 J whereas the WiFi uses 1.33 J (if already connected), which increases to 1.88 J considering the connection.

Concerning long-range connectivity, LoRaWAN power consumption varies widely with radio settings, such as the SF and the bandwidth, which affect the EBR. In LoRaWAN, the EBR ranges between 292 and 5469 bps. The overall energy for each uplink communication depends on the output power, the air-time (inverse of EBR), and the

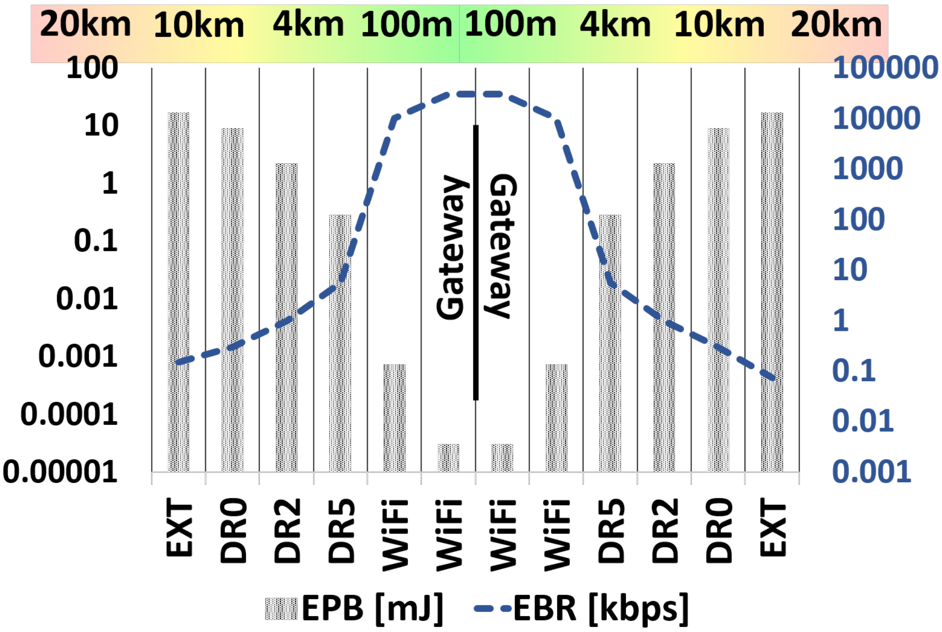


Figure 3.23: Energy per bit (EPB) and equivalent bit rate (EBR) varying the distance between the node and the gateway

receive windows open after RX1 and RX2 Delay. Table 3.9 presents the power consumption during one uplink, which includes the effective radio transmission, the RX1 Delay, and, finally, the receive window used to acquire the acknowledge (ACK). The payload size is 50 bytes, the maximum allowed for DR0, the CR, and the preamble symbols are respectively 1/4 and 8; finally, the CRC is disabled. In contrast with the limited range of WiFi and BT (tens to hundreds of meters, in ideal conditions), the LoRaWAN can reach 10 km of range, but the EPB increases significantly, and the EBR decreases. With the E-LoRaWAN, the energy consumption increases enormously, reaching 2.3 J per packet, but, on the other hand, it allows a range of more than 20 km. Table 3.10 and Fig. 3.23 show the EPB with respect to the distance between the gateway and the sensor node varying protocols and the radio configurations.

3.13 UWB: introduction

In recent years, UWB (Ultra-Wide-Band) technology has drawn relevant interest in the IoT community [133]. The development of UWB has led to a new era of short-range, high throughput wireless transmissions. Among various potential purposes, one of the most promising is in IoT [134–136], which needs both high-precision ranging capabilities and robust communications. Numerous research works in the literature indicate that UWB is one of the enabling technologies for wireless sensor networks [137–142]. In particular, impulse-radio-based UWB modulation has many inherent properties that are well suited to IoT applications. Indeed, it has potentially low complexity and low cost, with noise-like signal properties that create little interference to other systems. Moreover, it is resistant to severe multipath and jamming and has optimal time-domain resolution allowing for precise ToF estimation. Disparate UWB applications include locating of objects and environments, video surveillance, in-vehicle sensing, intrusion detection, outdoor sports tracking, monitoring of roads, bridges, and other civil infrastructure [143]. There have also been many reported systems to demonstrate the feasibility of UWB deployments for IoT applications, including radio modules and UWB SoCs and precision locating methods [144]. Several UWB-based sensor concepts have been designed both in the industrial and military domains, with particular importance on systems based on the IEEE 802.15.4a standard [145]. It provides predefined yet flexible PHY and MAC layers suitable for a wide variety of applications. Furthermore, it could operate concurrently with the ZigBee standard, a dominant technology in IoT systems.

In 2004, the IEEE established a standardization group, the IEEE 802.15.4a, with the mandate to design a new PHY layer for applications such as IoT and sensor networks. The 802.15.4a standard aims to provide an improved communications capacity to the 802.15.4-2006. It also grants ranging support to enable geolocation applications. One option of this standard is based on UWB techniques, namely, TH-IR. The IEEE first developed application scenarios, from which the requirements for the capabilities of the physical layer and channel models were deduced. In March 2005, a baseline proposal [146] was approved, and in the subsequent months, several subgroups developed the specifications of the modulation/coding schemes, concurrent channel access, ranging waveforms, and the MAC layer. On March 22, 2007, P802.15.4a was approved by the IEEE-SA Standards Board and was published in June 2007 [145].

Rapid developments in the area of intelligent autonomous vehicles over the past decade have seen their use extend to new fields, often in combination with other technologies to serve more complex purposes [147]. An emergent hot topic is to use UAVs to perform localization of nodes in WSNs [148], with further application to tracking first responders [149]. Knowing the precise location of a node, the UAV can fly to it for charging and data acquisition purposes [150, 151]. While data collection using a wireless interface poses an interesting challenge [152], autonomous navigation to within a few centimetres for the purposes of highly efficient inductive power transfer between a UAV and a device has not yet been practically achieved.

UWB is a promising technology for localization because it can perform distance measurements with accuracy down to 8 cm [153, 154]. However, its main disadvantage is its relatively high current consumption, i.e., up to 200 mA in the receive mode [155]. Indeed, many of the off-the-shelf UWB products are mains-powered or have extremely limited battery lifetimes, for example, in [156] and [157], where the battery lasts for only tens of minutes. On one hand, developing a localization system that can accurately estimate fixed node position can enable UAVs to move to unexplored areas where they may need to find their way to distributed wireless sensors. This mechanism can bring the drone relatively close to a sensor node to efficiently receive information, and with power transfer in mind, to recharge it [151]. Previous work also demonstrated the flexibility of UWB to achieve energy efficiency in an embedded system without losing accuracy [153]. This Section focuses on using UWB with hardware-enabled asynchronous communication to achieve energy-efficient wireless sensor operation in combination with UAVs. In particular, the UAV uses distance estimation using UWB with asynchronous mechanisms that allow the node to enable the radio transceiver only when the drone is close to it. This asynchronous duty-cycled UWB approach is under-explored; however, it is clear that the energy performance achievable may still be relatively poor due to the high current draw in the listening mode. Thus, a passive WURs can play an important role in developing an energy-efficient localization system that incorporates UWB to deliver centimeter accuracy.

3.14 WUR

In a typical sensor node, the radio transceiver is the most power-hungry component; therefore, the communication energy efficiency heavily impacts the average working time of these battery supplied devices. In previous works, such as [158], it has been demonstrated that aggressive duty cycling, turning the radio off and on periodically, significantly improves the lifetime of the network [159]. However, the duty cycling mechanism still has two side effects: the listening power consumption is not totally removed. Indeed, the transceiver needs to check the medium periodically with an intrinsic trade-off between latency and power consumption. In fact, the more the radio is powered off, the more energy is saved, but the latency will be higher. For this reason, asynchronous communication, which overcomes the latency/power trade-off, is considered one of the

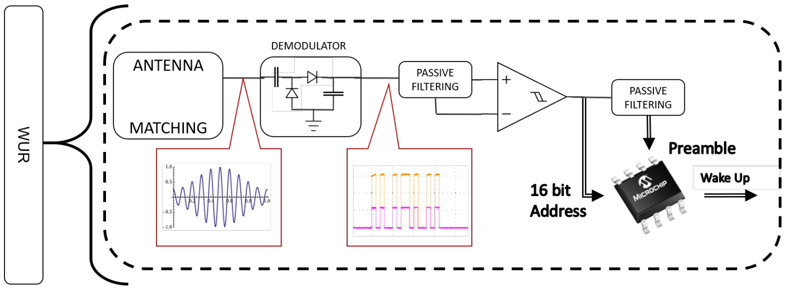


Figure 3.24: Wake Up Radio: the receiver schematic in which is highlighted the envelope demodulator, the high-frequency low-pass filter, the comparator and the MCU used to decode the address.

most efficient mechanisms for battery supplied sensor nodes [160]. In fact, with a WUR receiver, it is possible to decrease the idle-listening energy and achieve low latency communication, an essential factor in low-power devices. In this Section, a brief introduction of both receiver and transmitter is presented.

3.14.0.1 Receiver

Figure 3.24 shows the block diagram architecture of the WUR circuit, which can be used in conjunction with general sensor boards.

The receiver can decode only the OOK modulation, the simplest form of digital amplitude modulation in which each bit is represented as the presence or absence of a carrier wave. The WUR receiver architecture comprises four main blocks: the 868 MHz matching network, the fully passive envelope detector, the comparator, and a μW MCU used to decode the address (Figure 3.24). The matching network provides maximum power transfer between the antenna and the envelope detector by providing a stable impedance match with a return loss of -40 dB. It consists of an LC pi matching filter, whose component values are dimensioned based on the transmission carrier, the European ISM (Industrial, Scientific, and Medical frequency band) 868 MHz, and bandwidth 100 kHz.

The second block is the passive demodulator, which aims to recover the radio signal. Because WUR uses OOK modulation, the circuit consists of a passive envelope detector that discards the carrier frequency and its phase, only detecting the amplitude. The receiver makes use of a single-stage half-wave rectifier with series diodes, the $BAT15-04W$ RF Schottky diode pair from Infineon Technologies, which are optimized for frequencies up to 12 GHz. They offer a sensitivity of -56 dBm with the double diode schematic [161]. Once the signal is rectified, the digital bits of the received wake-up address, which follows an 8-bit preamble, are reconstructed by using an ultra-low-power comparator. To be robust at the input power changes, an adaptive threshold mechanism (Figure 3.24) is used, which keeps the comparator negative input at half of the input signal level. This enforces a limitation on the address compositions, where sequences longer than three ones or three zeros are forbidden. With this approach, the WUR uses the energy from the received signal for generating the threshold, thus reducing the static power consumption of the circuit rather than using a voltage divider. The TLV3701CDBVT is a comparator from Texas Instruments, which features a very low voltage offset of 250 μV and current consumption of 560 nA at 3.3 V. This component exploits the entire working range of the BAT15-04W diode. Aiming to keep as low as possible the power consumption, the WUR embeds an 8-bit microcontroller, the PIC12LF1552 from Microchip, which is selected for its ultra-low current consumption (20 nA in sleep mode), fast wake-up time (approximately 130 μs at 8 MHz), and an operating current of 30 μA/MHz, which is 30% lower than the STM32WB55RG. When the PIC12LF1552 detects a valid address, it wakes up the STM32WB55RG through an interrupt pin (Fig. 3.24), which enables the DW1000 and the ATWR ranging protocol.

The average power consumption of the WUR receiver is 2 μW, which reaches 82 μW during the address decoding. Hence, for each reception, the WUR needs 6 μJ considering a processing time of 24 ms (t_{beacon}).

3.14.0.2 Transmitter

The WUR transmitter aims to wake up the sensor node with as low latency as possible. It is, moreover, part of the drone payload and so must be lightweight. It includes a simple 8-bit MCU, the PIC16LF1824T39 from Microchip that has been chosen for its current consumption in active mode, only 96 μA at 3 V and, 16.5 mA in transmission mode at 10 dBm. The usage of the WUR transmitter is straightforward. Once powered up, the PIC16LF1824T39 continuously streams the wake-up packet with the destination address every 30 ms (P_{beacon}). The output power is 14 dBm. The transmission is done using 1 kbps data rate and a beacon containing a 16-bit address with 8-bit preamble.

Considering the transmitter output power and receiver sensitivity, the communication link budget is 70 dB. To evaluate the effective functionality of the WUR in the proposed framework, the path loss with both free space and multi-path model is calculated, taking into account a worst-case scenario with a -5 dB gain antenna. In Figure 3.25, simulations show the maximum coverage for each propagation model. In particular, Est. Flath Earth Loss (Figure 3.25) shows that the signal power goes below the WUR sensitivity after 10 m, while the free space model reaches this value over 40 m.

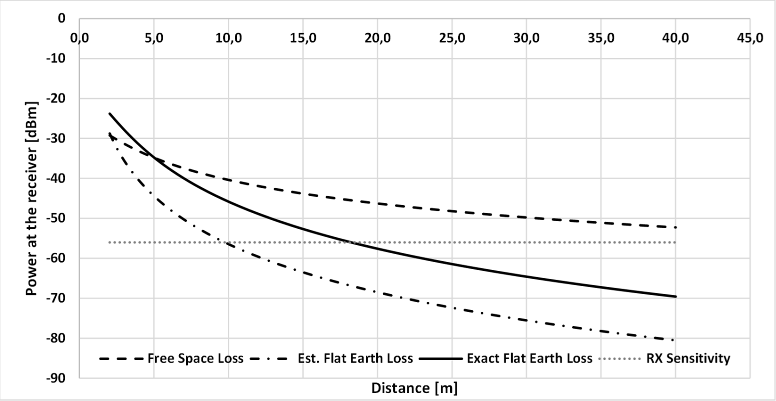


Figure 3.25: Wake Up Radio: path loss calculation with free space and multi-path model. TX: 14 dBm, RX sensitivity: -56 dBm, Antenna Gain: -5 dB, Connector loss: -1 dB

3.15 Ultra-Wide band

3.15.1 Decawave DW1000

The Decawave DW1000 [162] is a low power radio transceiver compliant with IEEE 802.15.4-2011 standard. It is a System on Chip (SoC) embedding a wideband radio front-end. It contains a receiver, a transmitter, and a digital back-end. A serial digital bus interfaces the SoC to the host processor. The radio transceiver supports six bands between 3.5 GHz and 6.5 GHz with two selectable data rates: 100 kbps and 6.8 Mbps.

The ultra-narrow electromagnetic impulse has a time width of 2 ns at a carrier of 3.9936 GHz, where the IEEE 802.15.4-2011 standard [155] gives the setting. The time width is proportional to the bandwidth (BW), which is 499.2 MHz. Hence, the internal clock is referenced to this frequency, providing a down-scaled 63.8976 GHz with a period of 15.65 ps. Due to the carrier-based impulse, the DW1000 physical layer can only generate +1 or -1 impulses with a bandwidth of 499.2 MHz at the selected carrier frequency. By grouping unique patterns of impulses, the link-layer obtains two main properties: periodic autocorrelation in coherent and non-coherent receivers, and low cross-correlation between colliding codes in the channel. The preamble symbols are drawn from a ternary alphabet -1, 0, +1 and have two different lengths, $Preamble_3$ and $Preamble_2$ that are respectively 1024 and 128 chips. The subscript number denotes the operating mode of the DW1000, which is always considered as 3 from this point (see the Decawave user manual for more details [162]). Inserting zero-valued chips between the elements, the ternary code is then spread. This spreading yield has a nominal pulse repetition (PRF) of 64 MHz, where each symbol has a period of 1 μs. During the

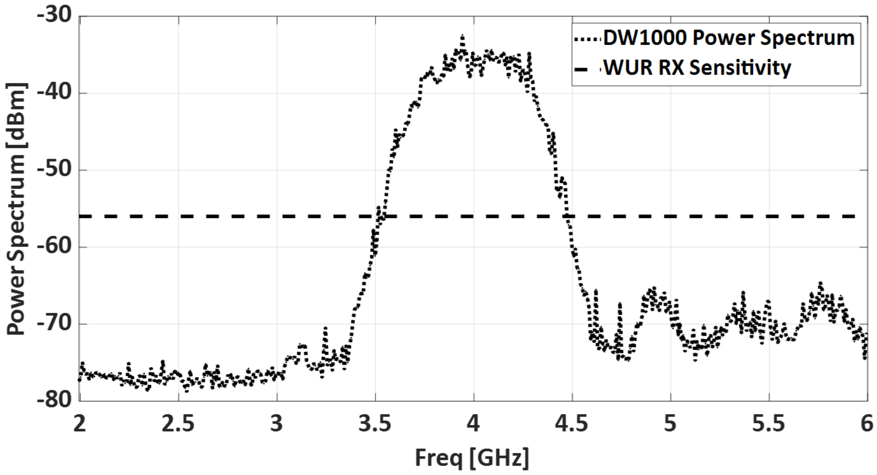


Figure 3.26: DW1000 power spectrum in mode 3; the horizontal line highlights WUR receiver sensitivity.

frame data parts, information bits are signaled by a modulation scheme defined as burst position modulation (BPM).

The DW1000 payload consists of symbols, where every single symbol carries two information bits [162]. Following, a symbol rate of 110 kbps is considered, denoted T_{sym}. It is composed of four slots of equal length, two burst slots detached by two guard intervals, used to prevent intersymbol interference. Each burst slot is subsequently split into eight sub-slots. The transmitter sends the burst in one of these, with a pseudo-random hop-position selection, providing multi-user interference rejection. It is relatively easy to understand that the majority of the channel time is empty during the transmission of a UWB packet, with a consequent reduction of the average transmission power that, in any case, is limited by the IEEE 802.15.4-2011 regulation [155]. In Fig. 3.26, the DW1000 power spectrum is plotted, where it should be noted that the maximum peak is around -35 dBm with an equivalent power spectral density of -110.8 dBm/Hz. This value is measured at the DW1000 power amplifier output, and thus it is higher than the radiated signal power from the antenna. Taking into consideration the ratio between the DW1000 output power and the WUR sensitivity, it is clear that making a WUR that uses the UWB radio-signal drastically reduces the communication range. Indeed, the budget link is only 21 dB. Due to the carrier high frequency and the reduced budget link, the communication range is only 10 cm. It is clear that this solution cannot be effectively deployed in most of the application scenarios, where an 868 MHz carrier is needed to augment the coverage.

3.16 Summary

This section has shown the most common and essential protocols used in IoT, proposing innovative and state of the art solutions to outperform and push the actual technology limitations. Moreover, the text studies explicitly and assess the power consumption, presented as Energy Per Bit (EPB) and/or energy per packet (EPP) in comparison with the coverage and the throughput (EBR). The presented results span between the common IoT standards such as WiFi, LoRa, LoRaWAN, Bluetooth, BLE, and NB-IoT also providing a detailed overview of emerging or custom technologies: UWB, ATWR, E-LoRaWAN, S-LoRaWAN, and Wake Up Radio (WUR).

Section 3.1 presents an analysis of the LoRaWAN protocol focused on channel throughput and issues associated with its scalability, one of the major concerns for this standard. Moreover, a reliable synchronization method is described and implemented, which is used to develop an S-LoRaWAN protocol suitable for low-cost and low-power IoT devices. This approach gives a theoretical 2× network throughput improvement. Still, in high traffic set-up, the measured gain is up to 5.8× with a demonstrated reduction of packet collisions of 26% in a real-life deployment with 24 nodes operating for hours. The overall overhead of the proposed S-LoRaWAN is only 8 bytes in the downlink packet, with little consequence in power consumption, which is a fundamental factor for battery operated devices. Finally, S-LoRaWAN does not require any change to the LoRaWAN software stack and can be deployed on top of unmodified LoRaWAN firmware.

Section 3.7 introduces the new NB-IoT and compares it with LoRaWAN for industrial application scenarios that require the transfer of a few bytes per day. The evaluation is based on experimental results obtained in-field, expecting a sensor node for crack measurements in civil structures [26]. Comparing both technologies shows experimental results in different coverage conditions, intending to assess the energy consumption, the estimated battery lifetime, and the packet loss. It shows that LoRaWAN outperforms NB-IoT in terms of energy consumption. In an application where buffering is not allowed, the LoRaWAN protocol increases the battery life up to 10× against NB-IoT: for Packet 3 scenario (36 Bytes payload), DR2 / M Coverage, NB-IoT EPB is 10× higher compared to LoRaWAN. However, NB-IoT is adequate for applications where information can be buffered on the node because each transmission energy is almost independent of the payload size. For example, in Packet 33 scenario (396 Bytes payload), DR2 / M Coverage, if high delivery latency is tolerable, NB-IoT EPB is 11× lower compared to LoRaWAN, due to the larger number of messages sent by LoRaWAN. Moreover, Section 3.7 verifies that T_{active} in the NB-IoT is heavily dependent on network coverage, as it grows up to 3× times passing from a "Good" (average RSSI of -80dBm) coverage to

Table 3.11: Final comparison between IoT protocols

Protocol	EBR	EPB	Range
LoRa EXT	147 bps	16.9 mJ	20 km
LoRa DR0	293 bps	8.87 mJ	10 km
NB-IoT Bad	-	2.7 mJ	10 km
LoRa DR2	977 bps	2.22 mJ	5 km
NB-IoT Medium	-	1.0 mJ	1 km
LoRa DR5	5.46 kbps	280 μJ	1 km
NB-IoT Good	-	0.9 mJ	100 m
UWB Mode 3	100 kbps	2.1 μJ	100 m
WiFi (UDP)	29.6 Mbps	30 nJ	20 m
UWB Mode 2	6 Mbps	44 nJ	20 m
Bluetooth	200 kbps	730 nJ	10 m

a "Bad" one (average RSSI of -130dBm). On the other hand, NB-IoT offers the highest QoS, which guarantees data delivery. This feature makes it a potential replacement to LoRaWAN in all the applications where the energy constraint is not an issue or when communication reliability is a crucial factor and good NB-IoT coverage is available.

Section 3.10 proposes a multi-protocol solution that aims to cover all the wireless transmission requirements for healthcare and IoT applications supporting WBAN networks and high-coverage through LoRaWAN. These low data rate protocols pair with a high throughput 22 Mbps WiFi connection, which allows the transfer of real-time raw data, such as images or multi-channel bio-signals. Moreover, Section 3.10 proposes E-LoRaWAN, a multi-hop solution that extends the LoRaWAN coverage for uplink and downlink, allowing continuous monitoring even in challenging environments or where it is not economically viable to deploy and maintain gateways. It studied the E-LoRaWAN limitation in terms of the maximum packet per day and the maximum number of out-of-range nodes per repeater, providing a simple way to deploy a custom IoT network. Compared to commercial products, the presented MultiRadio offers a diverse set of protocols satisfying all the possible requirements; with a 22 Mbps WiFi at 30 nJ per bit and the EPB equal to 730 nJ for the BT, the power consumption is reduced respectively by 2× and 5×. Moreover, E-LoRaWAN uses 2.7× less EPB on equal terms, allowing multi-hop communication that extends the network coverage area of 2×.

As conclusion of Chapter 3, Table 3.11 shows the performances of each aforementioned IoT protocols highlighting differences in terms of EBR, EPB and coverage. Proposed values are measured in real applications and realistic environments, such as consumer and industrial buildings other than in line of sight. The selection of the perfect technology for each specific application is challenging and is still an open issue in the IoT world. Indeed, it is not straightforward covering all the requirements, which are often asking for contradictory and inconsistent features, such as ultra-low power consumption and operativity in kilometers-range.

Chapter 4

Real-time distance estimation and data transfer

4.1 Two-way ranging

An asymmetric double-sided two-way ranging scheme [144] is implemented by Decawave SDK to determine the distance between two DW1000 [163]. Using physical and mathematical techniques to implement a clock with ps precision, the DW1000 can determine the radio Time of Flight (ToF). Assuming radio electromagnetic speed as the same as the light speed c, it is possible to use Eq. 4.1 to calculate the distance between two objects.

$$d = c \cdot ToF \tag{4.1}$$

In support of specific deployment scenarios, the two-way ranging benefits from being used in stand-alone mode without requiring complex infrastructure or fixed anchors [163]. The ATWR (Asymmetric double-sided Two-Way Ranging) is asymmetric because it needs two transmissions and one reception, and does not require equal and fixed reply times from each node. Its protocol is composed of three messages: Poll, Response, and Final. The maximum error is in the picosecond scale, even with low-cost 20 ppm frequency drift crystals, i.e., the worst-case specification [162], providing a theoretical error of approximately 2.2 mm. Each ATWR exchange consists of the UAV sending the Poll message, receiving the Response message, and then transmitting the Final message. The protocol sequence and the ToF formula are in Fig. 4.1 and Eq. 4.2

$$ToF = \frac{t_1 t_4 - t_2 t_3}{t_1 + t_2 + t_3 + t_4}. \tag{4.2}$$

Since t_2 and t_3 have a fixed and defined value, 800 μs with an error of ±15.65 ps, from t_1 and t_4 the microrontroller calculates the round trip time, which is two times the line of sight.

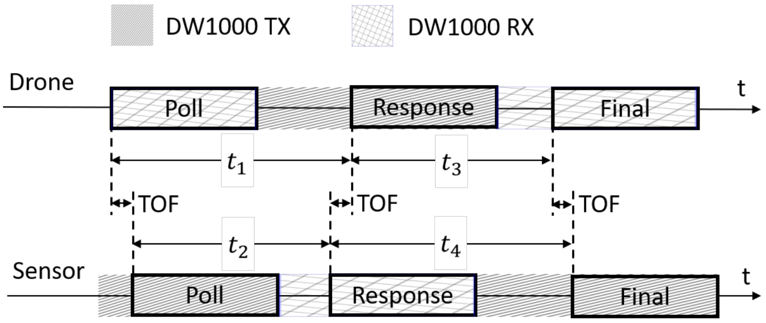


Figure 4.1: Asymmetric double-sided two-way ranging method. Sequences of packets for ToF estimation are shown.

4.2 ATWR Ranging Performance

The ATWR was tested under both static and dynamic conditions to verify the system accuracy and reliability. Stationary evaluations verified the distance estimation accuracy and the UWB minimum working range. The latter mainly depends on clock resolution and drift, affecting the ToF calculation [162]. On the other hand, the maximum ranging distance depends on the modulation settings, environment, and multi-path fading. Fig. 4.2 shows 21 different tests in $Mode(2)$ and $Mode(3)$ (Table 6.1) at 5 fixed distances: 10 cm, 20 cm, 90 cm, 3 m, and 5 m. The average and the standard deviation of the measured distance are generated from statistical analysis of over 75,000 points collected in a controlled environment. The ratio between the reference and measured distances is plotted in Fig. 4.2, where the central mark indicates the median, and the bottom and top edges of the boxes show the 25^{th} and 75^{th} percentiles. The whiskers continue to the most extreme data points not considered outliers, plotted using the o symbol. Fig. 4.2 shows that the UWB is not reliable in 10 cm and 20 cm range. Indeed, the measurement span reaches up to 100% and the error ratio in $Mode(2)$ and $Mode(3)$ is between 20% and 80%. UWB becomes reliable at the threshold of 30 cm. Although the variance is still high, the absolute error is 33 mm, corresponding to a bias of 11%. Between 50 cm and 5 m, the global performance improves significantly, and at 50 cm features an average bias of 5% and variance below 20%.

Twenty in-field evaluations were performed with a maximum speed of 60 km/h in addition to static analyses. The UWB behavior at varying speeds, compared with the results in Fig. 4.2, was examined studying 100,000 collected data points, where non-negligible packet loss due to fading and environmental noise was observed. The experienced data loss, observed with an average of 21% and a maximum of 80%, affects the ToF estimation and can cause abrupt changes and vertical edges/spikes, generating unpredictable behavior in the distance estimation. A non-linear digital filter technique,

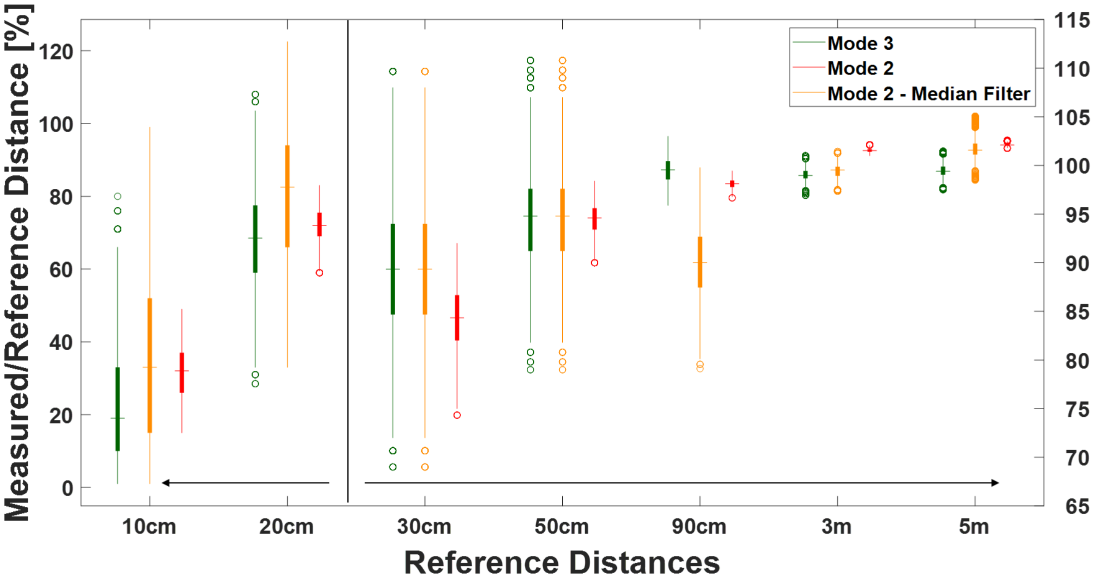


Figure 4.2: ATWR ranging performance in fixed positions. The graph shows the measurement variance of each test and the ratio between the reference and measured distance. The left part of the graph refers to the left scale, but the remaining refers to the right scale.

the 1D-median-filter, is applied to compensate the ToF estimation. As shown in Fig. 4.2, the median-filter with 21 entries (N) reduces the variance of a factor of three. Despite improvements in accuracy and noise reduction, this filter adds a time delay of $[int(N/2)+1]$ that must be considered while executing the position estimation algorithm. Moreover, it was noticed that the median-filter biases the 30 cm test, adding an error of 5%.

4.3 UWB: real payload throughout during data transfer

During in-field experiments, the payload throughput was extensively assessed in both $Mode(2)$ and $Mode(3)$, reaching an average value of 5.988 Mbps using a packet size of 1 kB. The test was performed in an open space with one-meter distance between two DW1000s devices, an environment with low electromagnetic noise. With a packet loss below 10% and a negligible protocol overhead (2 bytes), the achieved throughput reaches the DW1000 bitrate. During the data transfer, the equivalent energy per bit (EPB) is 44 nJ and 2.1 μJ respectively for $Mode(2)$ and $Mode(3)$. Hence, the 110 kbps setting reaches a broader coverage but requires more energy per bit, increasing the transmission cost.

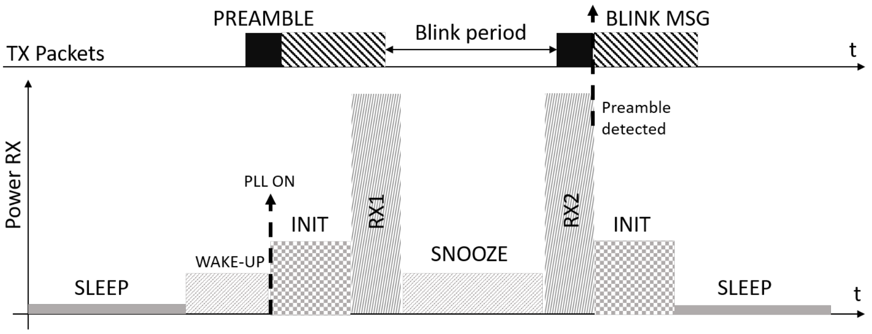


Figure 4.3: DW1000 low power listen procedure. The DW1000 generates an interrupt if a preamble sequence is detected in RX1 or RX2.

4.4 ATWR optimisation for energy efficiency

Concerning all DW1000 operation modes, the receiver mode is the most power hungry. In $Mode(2)$ and $Mode(3)$ it requires respectively 122 mA ($I_{2,rx}$) and 81 mA ($I_{3,rx}$) (with an external DC/DC converter), almost doubling the transmission current: 69 mA ($I_{2,tx}$) and 50 mA ($I_{3,tx}$). On the other hand, the sensor board needs to listen to incoming drone radio signals; thus, it must continuously monitor the ether. For battery supplied devices it is clear that a constant current of $I_{p,rx}$ is not sustainable, where the average current should be lower than 9 mA to achieve one month of operation.

An unique low-power listening (LPL) feature of the DW1000 [162] is evaluated in this Section; it aims to increase the battery lifetime at the cost of a duty-cycled operation, where the latency is not considered a priority. In low-power listening, the DW1000 wakes up periodically for a short time from the SLEEP state. The DW1000 automatically returns to SLEEP if no preamble is detected, following a procedure described in Fig 4.3. Once a preamble is marked, the MCU receives an interrupt and configures the DW1000 in continuous listening mode, enabling the ATWR ranging. However, the transmitter has to send sufficient data to wake up the sensor node, ensuring a non-null probability to hit the short receive window, Fig. 4.3 RX1 and RX2. Essentially, the transmitter has to stream for a time >= 1 window time width. In practice, this is done by sending the same beacon repeatedly. The DW1000 also includes the ability to do a two-phase listening to increase the probability of hitting a preamble and to decrease the wake-up latency. The two-phase listening is composed of two reception windows (Fig. 4.3 RX1 and RX2), spaced out by a short sleep period, which is defined as SNOOZE time in Fig. 4.3. It is set to ensure that if the first listen hits a message, missing the preamble, then the next window will see it. The SNOOZE time can be configured as the Blink period, 33 ms, assuming that the ATWR procedure is executed at 30 Hz. Consequently, a time width >= 33 ms must be used for RX1 and RX2. An empirical value of 50 ms is selected in this evaluation to increase the detection probability in real scenarios.

To calculate the duty cycle supported in the application scenario, the maximum wake up latency should be established. During this time, at least one couple of RX1/RX2 must be opened. However, a maximum packet loss (PL) of 80% is noticed from the in-filed test of the UWB, a detrimental factor that must be considered programming the wake-up sequence. Using the packet loss probability, it is useful to calculate the average number of transmissions (Eq. 4.3) required to receive a wake-up packet.

$$N_{Blink} = \sum_{i=0}^{\infty} PL^i = \frac{1}{1 - PL}, \tag{4.3}$$

Furthermore, considering the effective N_{Blink} that must be performed in a real deployment, the low power listening period Φ is calculated as shown in Eq. 4.4 and the corresponding sleep time in Eq. 4.5, where t_{init} is equal to 3 ms.

$$\Phi = \frac{\hat{d}}{|v|\, ceil\left(\frac{N_{Blink}}{2}\right)}, \tag{4.4}$$

$$t_{sleep} = \Phi - t_{RX1} - t_{RX2} - t_{init} - t_{SNOOZE}. \tag{4.5}$$

For example, with a t_{sleep} of 1.63 s, considering Φ equal to 1.75 s, the corresponding DC is 7%. Alternatively, Φ and t_{sleep} can be 15 s and 14.86 s with a DC of 1%, aiming to decrease the average power consumption.

4.5 ATWR and WUR: an asynchronous approach

This section presents a fusion approach between ATWR and WUR technology. The resulting radio protocol is straightforward: the sensor node is in deep-sleep mode with all the peripherals powered off. The only active component is the WUR receiver, reference in Fig. 4.4 labeled as *node in shutdown*. Approaching the targeted destination, the drone needs to send wake up beacons (Fig. 4.4 WUR TX) using the PIC16LF1824T39 in the 868 MHz band. These packets are streamed starting from the maximum UWB range, 100 m, which aims to enable the ATWR. Immediately after, the DW1000 on the drone sends the Blink message (Fig. 4.4) to check if the equivalent component on the sensor side is ready. The Blink message has a duration of 15 ms. Once the WUR beacon is received, the WUR module enables the MCU, which configures the UWB transceiver in a period equal to t_{init} (Fig. 4.4). At the next Blink transmission, the sensor node will correctly start the ATWR ranging protocol (Fig. 4.4). Considering the path loss calculation shown in Fig. 3.25, the WUR beacon should be received on average between 10 m and 40 m.

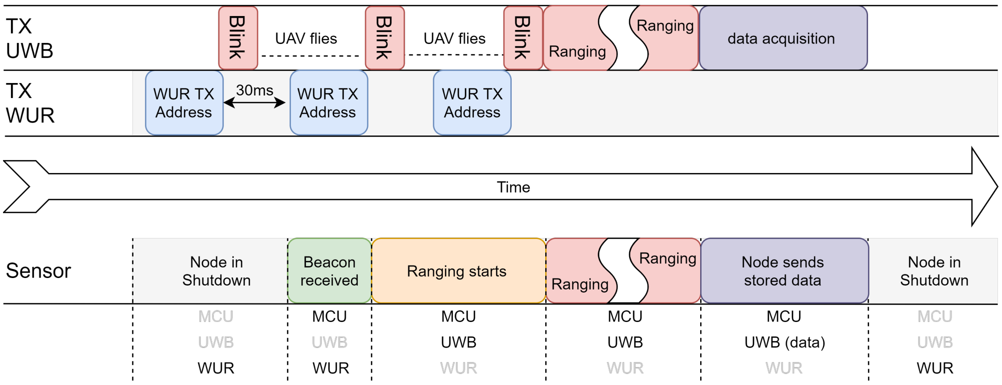


Figure 4.4: Asynchronous ranging protocol. UWB and WUR fusion for low power and low latency distance estimation.

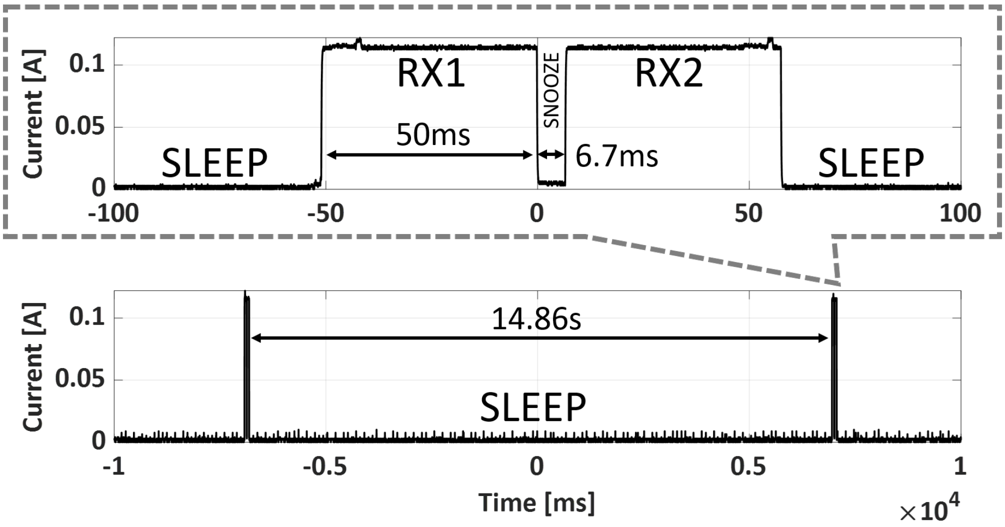


Figure 4.5: Top plot: Measured current consumption of the DW1000 low power listening mechanism. Bottom plot: Average current consumption with the SLEEP time.

4.6 UWB: Experimental Evaluation

To test the UWB low power listening capabilities, the DW1000 is programmed with settings described in Section 4.4 at 1% DC. Considering the configuration in $Mode(3)$, an average power consumption of 4 mW is expected, which includes only the DW1000. In Fig. 4.5 is plotted the current consumption of the low power listening mechanism (see Fig. 4.3). As expected, the receive window time, RX1 and RX2, is 50 ms, and the sleep time is 14.86 s, as well as the peak current, which is 120 mA. These values reflect the analytical results described in Section 4.4. The average power in low power listening mode is equal to 4 mW with the 1% duty cycle setting. Instead, the SNOOZE time is only 6.7 ms, i.e., 5 times less than the configured value. This nonconformity does not compromise the functionality of the system as well as the average power consumption, but it decreases the probability to hit the second receive window since the transmitting Blink period is not aligned with the SNOOZE time. However, this issue can be resolved by optimizing the DW1000 driver library, which casts the SNOOZE time at a maximum value of 7 ms. Finally, the average power consumption of the ATWR, resulting in a continuous 400 mW, heavily affects the overall battery life. Indeed, for instance, a 1 Hz sensor reading needs just 70 μW.

4.6.1 WUR: in-field test

A quantitative analysis of the performance of the WUR receiver was performed in terms of range and sensitivity. The WUR transmitter was configured to continuously send a 16-bit OOK address every 30 ms at the power of 14 dBm. The WUR receiver, which is placed on the ground, was programmed so that it lights an LED whenever a message is detected. In this range evaluation experiment, the transmitter firstly moves far outside of the WUR range. It starts approaching the sensor node point (therefore the WUR receiver location) from four different directions, as presented in Fig. 4.6. The WUR range was analyzed by observing the TX-WUR reported distance when the WUR led turns on. Both receiver and transmitter are equipped with a -2.3 dBi SMA antenna. This procedure was repeated 40 times, measuring a maximum communication distance of 15 meters. This result is consistent with Fig. 3.25 in which it estimates a communication range between 10 m to 17 m using Earth Loss models (Fig. 3.25). Due to the moving parts and the reflected waves generated in real environments, the packet loss cannot be considered negligible; hence it was measured at varying planar distance by setting the flight altitude at 2 m. Between 0 to 10 meters, the wake-up probability is above 90% with static objects and above 80% during the flight (Fig. 4.6). Beyond the 10 meters threshold, the link quality decreases. At 10 meters, the packet loss is 80%, which further increases to 90% at 12 m.

These results fully support the asynchronous ranging protocol (Fig. 4.4), providing high reliability within 10 m range.

The same transmitter was used to perform a sensitivity evaluation, connecting it to the WUR receiver through an adjustable RF attenuator. The transmit power was 0 dB. A logic analyzer was connected to the wake-up pin of the WUR. This pin gives a 1 ms pulse when the correct wake-up message is detected. Increasing the attenuation, results show that below -45 dB more than 10% of the messages are lost.

The measured current consumption of the WUR module is 716 nA at 3.3 V.

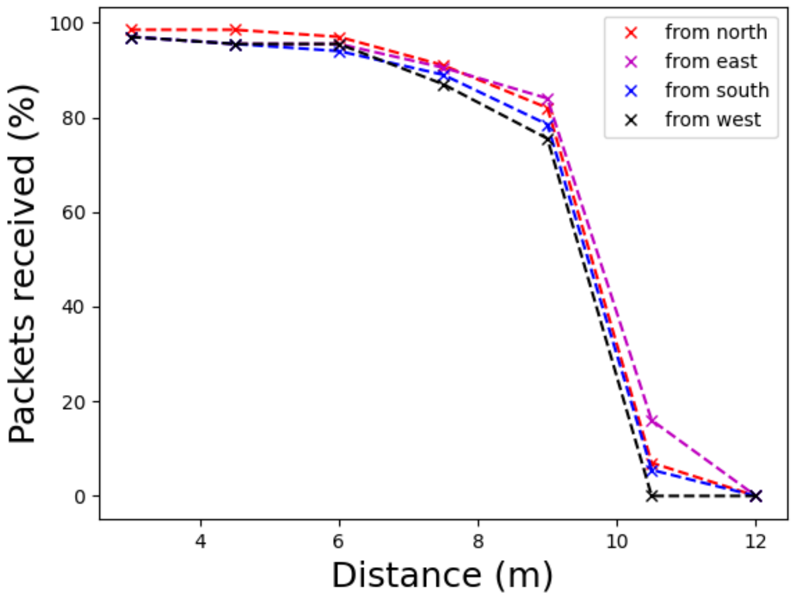


Figure 4.6: WUR packet delivery ration varying the distance and the angle of arrival. The drone is placed 2 m above ground.

4.6.2 LPL and WUR: in-field comparison

In the worst case, in terms of power consumption, the LPL setup can be woken up at the first receive window (Fig. 4.3 - RX1). This possibility is not negligible; indeed, considering N_{Blink} equal to 5, the first window probability is 20%. In the proposed evaluation, the sensor node needs 28.8 J for the LPL implementation without the WUR asynchronous wake-up.

In Fig. 4.7, the power consumption profile of the sensor board is plotted. The initial current spike highlights the WUR receiver operation, which does not correctly decode the beacon at the first attempt. After 30 ms, a new beacon is received and successfully decoded. Hence, the WUR circuit wakes up the MCU, which, after an initial configuration time, starts the ATWR ranging protocol. With this approach, the energy to support the ATWR is halved, enabling the expensive UWB protocol only when it is really needed. On average, using the same period, the measured energy is 11 J, 2.6× less than the LPL approach.

4.7 Summary

This Chapter presents a new open platform to enable efficient localization for wireless data and power transfer between UAVs and static wireless sensor devices. It experimentally verifies the effective throughput of the DW1000 in $Mode(2)$, which reaches a continuous speed of 6 Mbps. Moreover, it examines both low power listening UWB and wake-up radio approaches, providing models and measurements to estimate power consumption in non-ranging states and during the approach phase. This section effectively demonstrates both solutions with accurate in field-test comparisons, achieving an average energy reduction of 62% using the WUR approach, which deviates to ∼50% in challenging windy conditions. In these conditions, the WUR performance fully supports the precision localization, with a packet delivery ratio above 80% within 10 m range and a maximum latency of 30 ms. In proposed experiments, the maximum coverage is achieved between 2 m and 5 m altitude, at which the UAV should move during wake-up triggering to maximize the system reliability. The raw data from the field experiments reported in this section is publicly available on GitHub. They demonstrate the effectiveness of the framework, fusing the benefits of asynchronous WUR and UWB distance measurement. Finally, concerning the sensor node sleep current, it shows that the WUR solution reduces the average power up to 2000× and 14000× with respect to 1% and 7% in LPL mode. The operational lifetime of the *sensor* is thus significantly extended.

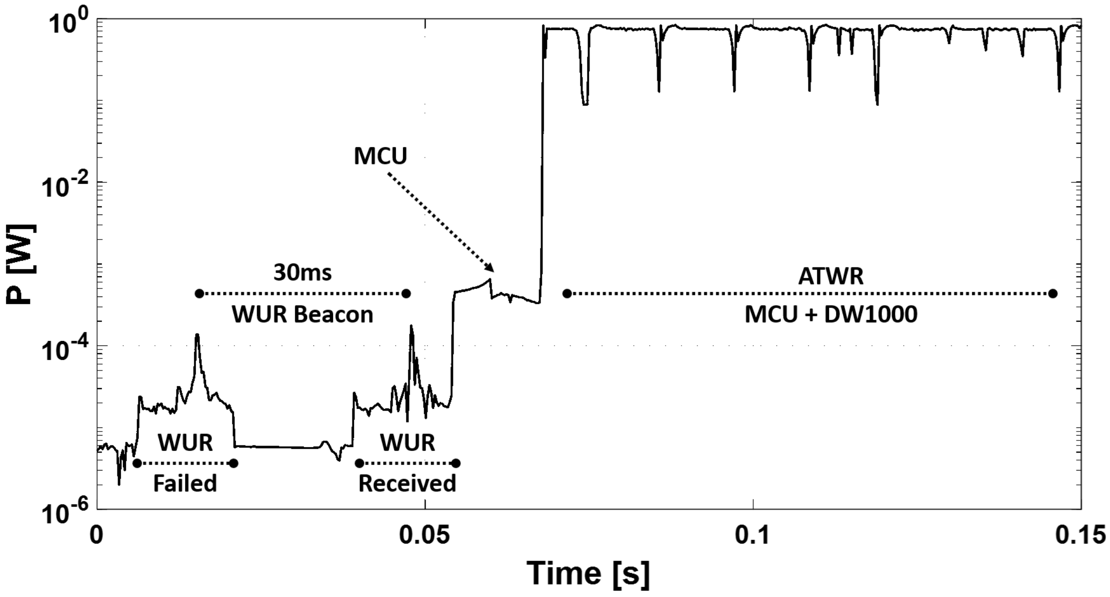


Figure 4.7: Power consumption profile of the sensor node. In this specific case, the first received WUR beacon was corrupted, and the receiver discarded the wake up sequence.

Chapter 5

Sensors for IoT: from scratch to a plug and play product

After introducing IoT features and limitations and a comprehensive study of the most used wireless protocols, this dissertation explores the design of a transducer. Sensing is a core block of IoT devices, and it is often the key component to realise an industrial or user system. Hence this subject is tackled not only describing and evaluating commercial off the shelf sensors, but providing a complete development pipeline, from scratch to a plug and play product, of a magnetoelectric transducer. The following chapters present an innovative and cost-effective sensor that measures and tracks cracks in concrete and other construction materials, proposing the structural health monitoring as application scenario. The final device (Section 6.1) combines a microprocessor with LoRaWAN wireless communication (Section 3.1) and a solar energy harvesting, allowing long-term remote monitoring with easy plug and play installation.

5.1 SHM: Introduction and use case

Structural health monitoring systems are becoming increasingly widespread and are, in some cases, mandated by law. There are more than 1 million bridges in Italy, and most of them were built between the 1950s and 1970s. Unfortunately, today, there are a limited number of inspectors available to control their status, and, in the last ten years, numerous victims were caused by bridges collapsed in busy roads. A similar situation happens for public buildings, where accurate and periodic monitoring of health conditions is lacking. Traditional inspections are subject to human error, and their interpretation relies on the inspectors judgment. Moreover, the evolution in buildings construction through the design and realization of increasingly complex structures with new materials and techniques requires tracking the structural integrity over many years [164]. A major factor limiting such diffusion is the lack of low-cost, low-power sensor nodes that can be deployed on a large number in hard-to-reach areas while providing high-quality and precise measurements.

The new concept of Autonomous Structural Health Monitoring (ASHM) is based on methodologies that do not require expensive instrumentation to provide the necessary information with the required accuracy. Some methods need accurate measurements done by experts and repeated regularly after a medium-long time interval. For example, the work in [165] uses lasers scanners to construct a 3D model of the buildings, that is later compared with a baseline to detect actual alterations of the environment under measurement. Continuous monitoring and frequent building evaluations allow for dynamic analysis of specific civil structures such as bridges or skyscrapers, [165, 166]. Being wired, these solutions still present a high cost due to the installation and use of piezoelectric sensors. Usually, in all these cases, the structure response analysis is calculated through automatic tools that run directly in remote servers or in a cloud, which retrieve data directly from a database. The SHM output is determined by processing the data acquired from the buildings when they are stressed by external mechanical excitations, such as vehicle traffic, hearth quakes, weather phenomena or wind, big loads, or by the use of mechanical shakers. Low-cost vibration ASHM devices and structural crack sensors are still a challenge, only a few examples are present in literature, and they are still in the first prototyping stage [167]. In [166], Addabbo and Fort present a new ASHM method for crack width measurement, which is based on a permanent magnet and a hall-effect sensor mounted in a 3D printing holder. Their paper shows a low-cost solution characterized by a considerable sensitivity ($> 10\ \mu m$), low power consumption, and some low-complexity electronics. Efficient energy harvesting techniques, combined with hardware that looks explicitly for ultra-low-power sensing, allow developing systems that can operate virtually forever [168]. Besides, wireless sensor networks are a promising technology for ASHM applications. Indeed, recent works have deployed similar autonomous solutions in different environments. However, the sensor cost is still so high in these works, and the wireless communication range is not enough for plug and play applications where the sensor could be anywhere.

New techniques for continuous and autonomous structural health monitoring are necessary to assist human inspectors with periodic data and improve the building assessment. The SHM process involves observing a structure over its life using one or more sensing elements, which are used for extracting relevant features. Traditional deployments for structural health monitoring are expensive ($> 1\ k€$), as high accuracy and sensitivity are required to the sensor and need to be guaranteed up to 10 years without manual calibration [26]. The cost of such apparatus often limits civil engineers and building companies, and they can only deploy a limited number of sensors in few positions. Unfortunately, to fully understand the dynamic movements and building stability, information coming from several spatial locations is needed [167]. Moreover, most of the products on the market are designed for short-term measurement campaigns where operators are supposed to monitor the measurement process.

The following chapters present an innovative and cost-effective transducer that measures and tracks cracks in concrete and other construction materials. The final device, presented in Section 6.1, combines a microprocessor with LoRaWAN wireless communication and solar energy harvester, allowing long-term remote monitoring with easy plug and play installation. Experimental results demonstrate that it can achieve about 1 μm accuracy , with stable measurements across a $-15-65°C$ temperature range. The instrument is based on two permanents magnets and a hall-effect sensor, characterized by a very competitive sensitivity at a lower price. Moreover, combining effective power management with a high-efficiency solar energy harvester (Section 6.1), it can guarantee more than 10 years of operational life, with 10 measurements per day. Finally, to assess measurement precision and reliability, an in-depth study of low-cost plastic materials for the sensor enclosure was performed verifying the performance over a wide range of operating temperatures, without the need for exotic and expensive materials.

5.2 Crackmeter design

The sensor presented in this paper is a low-cost displacement transducer based on permanent magnets facing each other with opposite polarity, a specific configuration developed in conjunction with STMicroelectronics, Sacertis S.r.l.[1], and Rosa Micro S.r.l.[2] The magnets (FE-S-20-03) are commercial cylindrical ferrite 20×3 mm, and a magnetisation of 0.40-0.41 T or 4000-4100 G. They offer excellent temperature stability of the magnetic field, in the range -60 *to* $250°C$, allowing the operation in different environments without changing their characteristics over the entire sensor life. Thanks to the geometric configuration of the two magnets, the combined magnetic flux B is highly linear with distance. The graph in Fig. 5.1 shows that in the range ±5 mm, the curve can be approximated to a linear response with a coefficient of determination R2 = 0.999; this result allows the use of a commercial and inexpensive Hall sensor guaranteeing a high output sensitivity response. The magnetic flux of two facing permanent magnets follows the physical behavior described in Eq. 5.1, Eq. 5.2, and Eq. 5.3, where D is the

---

[1] http://www.sacertis.com

[2] https://www.rosagroup.com

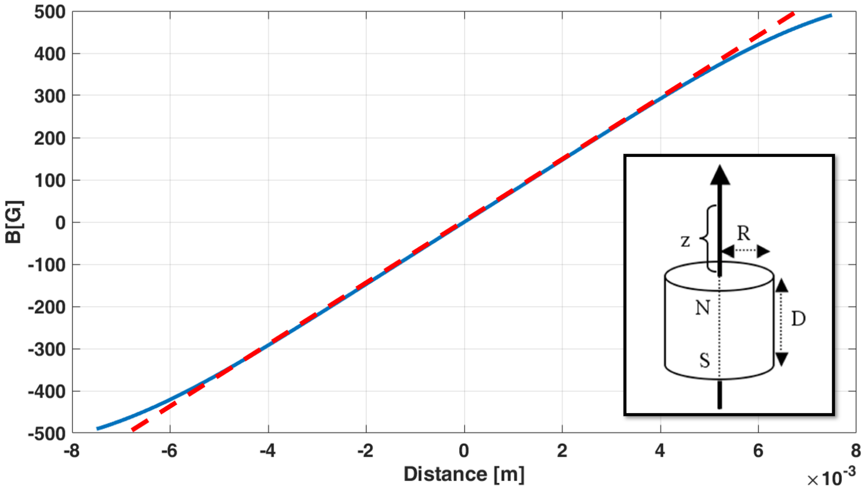


Figure 5.1: Combined magnetic flux B between two magnets.

cylinder thickness, R is its radius and z is the distance from the magnet. Lastly, B_r is a measurement for the magnetic induction or magnetic flux density.

$$B_{left}(z) = \frac{B_r}{2}\left(\frac{(D+z)}{\sqrt{R^2+(D+z)^2)}} - \frac{z}{\sqrt{R^2+z^2}}\right), \tag{5.1}$$

$$B_{right}(z) = \frac{B_r}{2}\left(\frac{(D+z)}{\sqrt{R^2+(D+z)^2)}} - \frac{z}{\sqrt{R^2+z^2}}\right). \tag{5.2}$$

$$B(z) = B_{left}(z) - B_{right}(z). \tag{5.3}$$

The Allegro A1319 linear Hall-effect sensor IC has been designed specifically to meet high accuracy and low power requirements. This stable-temperature device is available in both surface-mount and through-hole packages. The accuracy of the device is enhanced via end-of-line optimization. Each device features non-volatile memory to optimize the device sensitivity and the quiescent voltage output for a given application or circuit. This ratiometric Hall-effect sensor ICs provide a voltage output that is proportional to the applied magnetic field. The quiescent voltage output is adjusted around 50% of the supply voltage. The A1319 BiCMOS monolithic circuit integrates a Hall element, temperature-compensating circuitry to reduce the Hall element intrinsic sensitivity drift, a high-gain small-signal amplifier a clamped low-impedance output stage, and a dynamic offset cancellation technique. In Table 5.1, some fundamentals parameters of the Allegro sensor are outlined.

The two cylindrical ferrite magnets are distant 15 mm providing a theoretical measurement range of 12 mm. The last two millimeters on both sides cannot be used due to B non-linear behavior, which would affect the transducer sensitivity; see Fig. 5.1 as a reference. The Allegro A1319 is held by a non-ferromagnetic material between two magnets, which in this design is a plastic support. *Support 1*, in Fig. 5.2 is connected on the left side of the crack, it contains the electronic circuit and the Hall effect sensor.

Table 5.1: Allegro A1319 measured performance.

Parameter	Value
Sensitivity	5 mV/G
DC Output Resistance	$< 1\ \Omega$
Quiescent voltage output	1.65 V
Spectral noise density (0 - 100 Hz)	$15\ \mu V/\sqrt{Hz}$
PSRR (0 - 1 kHz)	5
PSRR (1 kHz - 100 kHz)	> 1000

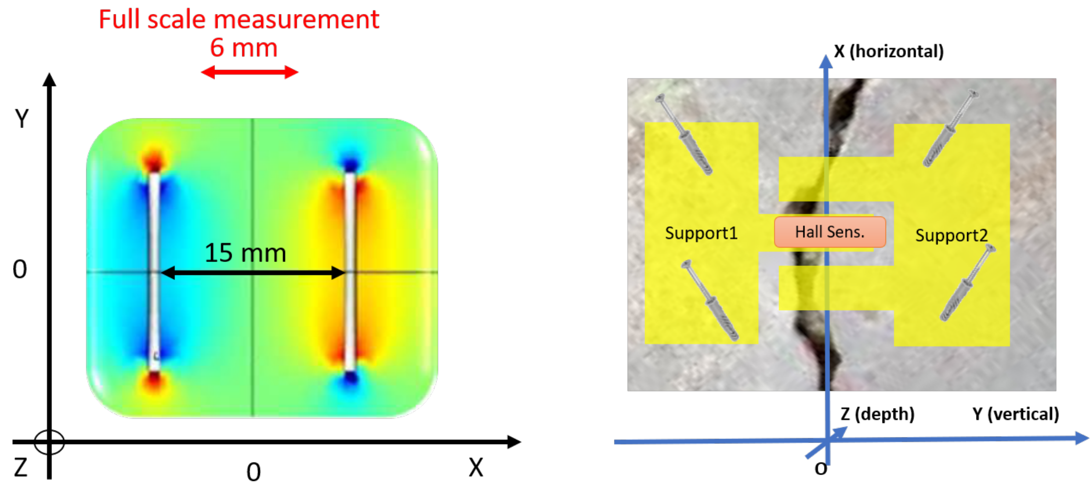


Figure 5.2: Magnets displacements and the corresponding holder position with respect to the crack.

Support 2, indeed, holds the magnets and is connected on the right side of the crack. It does not include active components and is used as a reference point for *Support 1*. Hence the transducer here presented measures the distance between *Support 1* and *Support 2*, which, in turn, are fastened to the concrete structure among crack strips. Both sides are free to move in three-dimensional space, but the distance estimation is performed only on the X-axis. For these reasons, movements in another direction should not change the transducer output, avoiding to degrade its performance.

Before making the first prototype, a complete assessment of the aforementioned transducer was performed to check the sensor sensibility from temperature and ageing and tilting variations. All the simulations in Section 5.2.1 have been published by courtesy of STMicroelectronics.

5.2.1 Crackmeter system validation

The following simulation results consider magnets with a diameter of 20 mm and thickness of 3 mm, while the fixed face-to-face distance is 15 mm. The Hall effect sensor is considered as a dot with zero spatial dimension. These tests aim to show the transducer behavior on the measurement axes (X) varying all the possible parameters, such as other axes (i.e., Y and Z), tilting between magnet face and the Hall sensor, and the magnet ageing or corruption due to external events. The reference point for all the axes is at the center of the magnet circumference and at the middle of the magnet distance, i.e. at 7.5 mm from each face. Compared with the real deployment, simulation axes are flipped, and the corresponding measure line belongs at the Z line. All the simulations span between ± 3 *mm* from the origin and between 0° to 10° for tilting.

5.2.1.1 System validation along X

The simulation results in Fig. 5.3(a) propose the measurement linearity and errors along the sensitivity axes of the Allegro A1319 (Z). To evaluate the dependency from X and Y, a trial of eight positions shows the sensor behavior varying only X position or both axes. When the lateral misalignment is zero, the plotted error displays the sensor non-linearity deviation in ideal conditions, which reaches a maximum of 5 μm at ± 2.5 *mm*. In the worst conditions, the maximum error is 26 μm and 37 μm varying X or X and Y, respectively. In both states, the maximum is reached at the highest lateral misalignment of 3 mm. Lastly, shrinking the measurement range between ± 1 *mm* the linearity error is < 1 μm.

Left plots of Fig. 5.3(a) show the full magnetic field span, which does not saturate the measurement range of the Allegro A1319 in the range ± 200 *G*.

5.2.1.2 System validation along X and Y

The simulation results in Fig. 5.3(b) propose the measurement linearity and errors along the sensitivity axes X, which is orthogonal to the Allegro A1319 measurement direction. Fig. 5.3(b) shows that, for future developments, using a two-dimension Hall sensor can enable a two-dimensional crackmeter; indeed, the magnetic field behavior along X is similar to Z. At the lateral misalignment equal to zero, the non-linearity deviation in ideal conditions reaches a maximum of 5 μm at ± 2.5 *mm*. In the worst conditions, the maximum error is 26 μm and 47 μm varying respectively Y or Z. In both states, the peak is reached at the highest lateral misalignment of 3 mm.

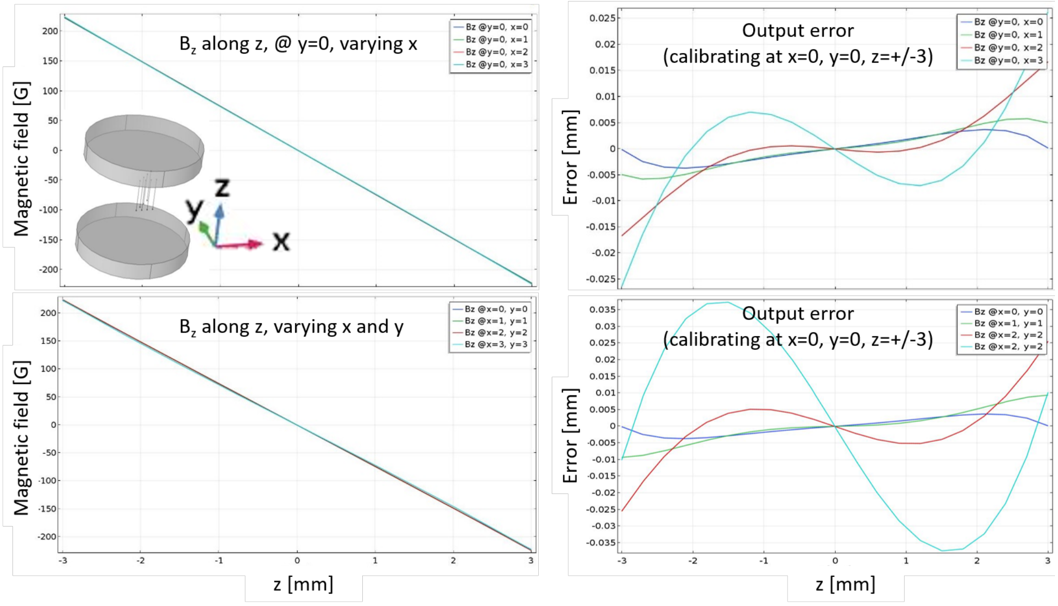
Next sections and the final deployment focus on one measurement direction along Z (for simulations) and X (for in-field tests). However, it is possible to exploit these results for future investigations or new commercial products.

5.2.1.3 System validation tilting magnets with respect to Z

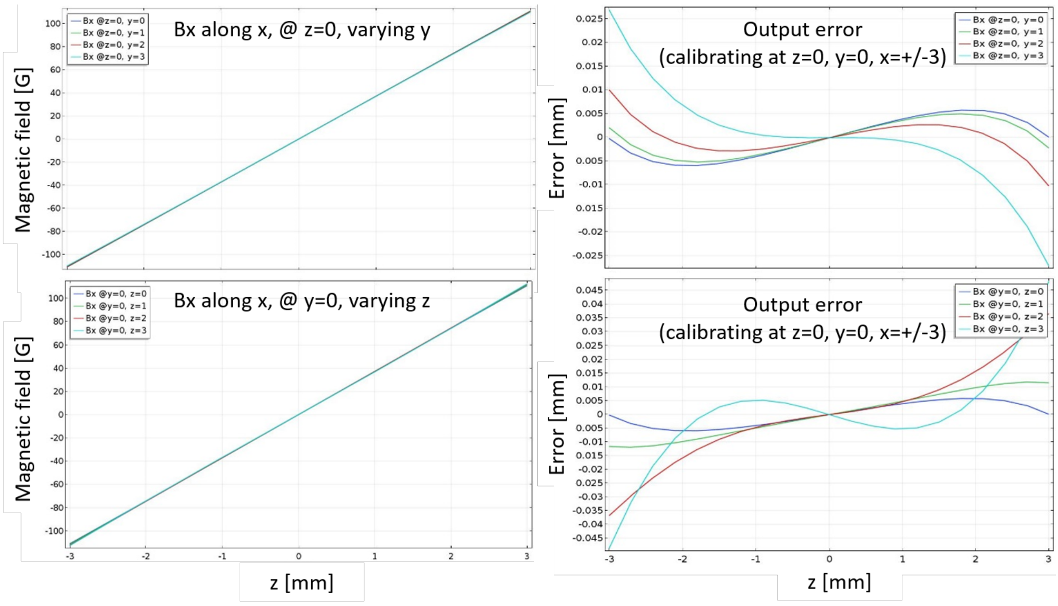
The simulation results in Fig. 5.4(a) present the measurement error along the sensitivity axes of the Allegro A1319 (Z) varying the magnets angle (α) with the latter. A trial of six angles, between 0° and 10° shows the sensor behavior changing α. When the angle is zero, the plotted error complies with the sensor non-linearity deviation detected in Fig. 5.3(a). The tilting generates the most destructive effects on the crackmeter, reaching a maximum error of 140 μm @ 3 mm. However, the error increases linearly with the distance from the origin containing the amplitude below 40 μm in ± 1 mm range. So, it is clear that aiming to reach a micro-meter sensitivity, the sensor enclosure should avoid as much as possible an angular misalignment between magnets and the A1319. Compared to lateral misalignment, which could be compensated using orthogonal lines of sense, it is difficult to overcome technical difficulties when tilting axes; indeed, it will affect all the sensing direction, making it hard to balance the inserted error.

5.2.1.4 System validation sweeping magnet parameters

To validate the sensor behavior during its entire lifetime, Fig. 5.4(b) shows some of the possible effects coming from aging and degradation, such as a variation on magnet height and radius due to corrosion and B_r decrease. Sweep simulations on height and radius consider a 1% variation, reaching a maximum error of 80 μm, while a 1% alteration on Br generates up to 35 μm variability. These results highlight the importance of a robust enclosure covering and protecting the magnets to avoid mechanical variations due to external events, such as hits, cracks, and dirties. Lastly, in Fig. 5.4(b), B_r varies only on one magnet, while, in a real deployment, this parameter should degrade equally on both objects since they are subjected to the same environmental conditions.

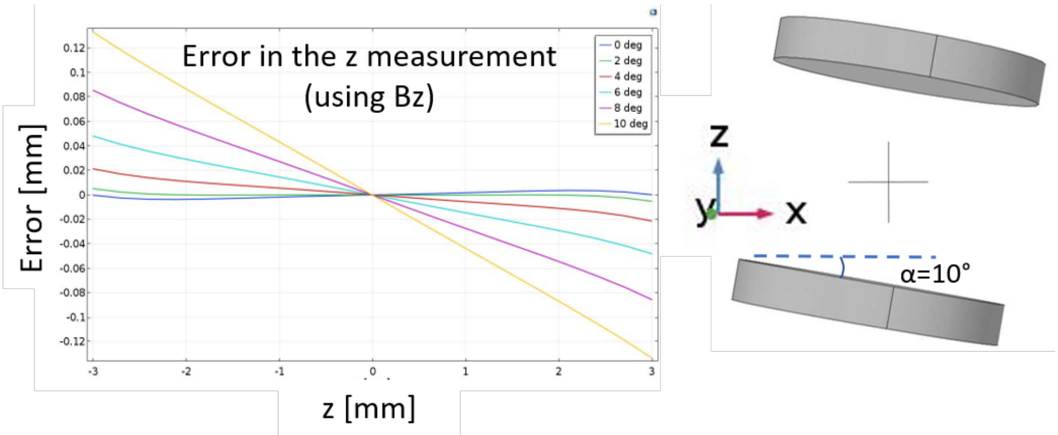


(a) Simulation results of the magnetic field along Z, right plots show the normalised errors varying Y and X.

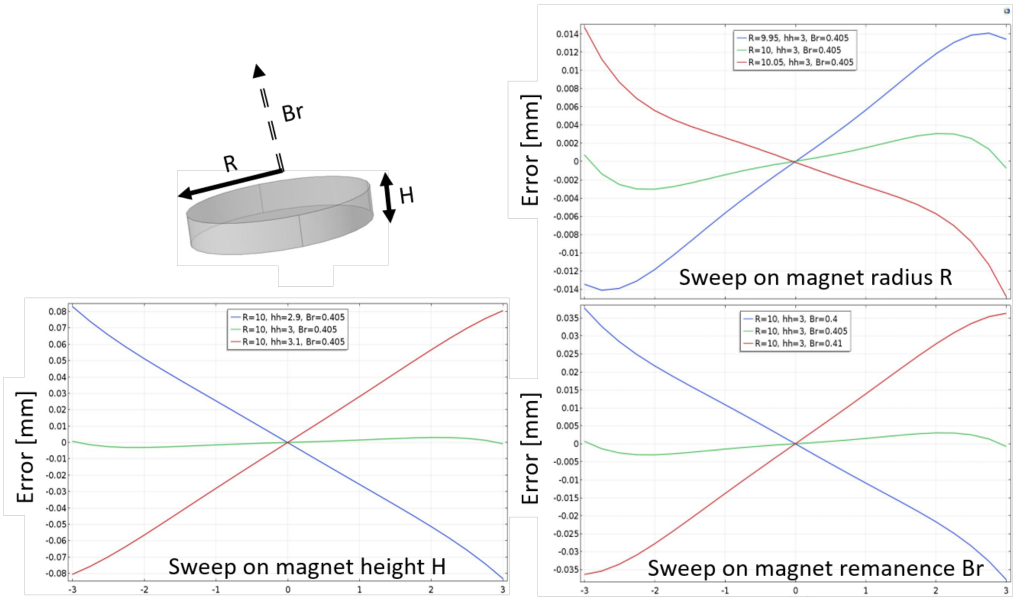


(b) Simulation results of the magnetic field along X, right plots shows the normalised errors varying Z and X.

Figure 5.3



(a) Simulation results of the magnetic field error along Z tilting the magnet with respect to the sensitivity axes of the Hall effect sensor.



(b) Simulation results of the magnetic field error sweeping the magnet height, remanence, and radius.

Figure 5.4

5.3 Crackmeter enclosure

Magnets and the Allegro A1319 must be kept in the right position, and also, the transducer needs to be solid with the wall and the crack under control. Hence a robust support is required, which must hold the components and, at the same time, shift along X, Y, Z axes following the crack movements.

The sensor plastic case is designed to minimize size with an appropriate mechanical rigidity to avoid undermining the overall output sensitivity. Performances must be stable in temperature and for the entire life of the system, that can be $>$ 10 years. Several plastics have been considered, and the best compromise between cost and performance has been found in polycarbonate (PC as per DIN7728). Polycarbonate is a thermoplastic polymer obtained from carbonic acid. The selected PC is GF30, a polycarbonate loaded with 30% glass fibers, showing greater strength, rigidity, creep resistance, and dimensional stability than unreinforced PC. In addition to good resistance to fatigue, these properties make the glass-filled polycarbonate a suitable material to be used to encapsulate devices that remain exposed for a long time to high static loads and high temperatures. Moreover, its thermal expansion is in the lower level of the corresponding categories ($5\ 10^{-5} K^{-1}$), and surface resistivity is $10^{14}\ \Omega$. Unfortunately, PC cannot be 3D printed, a significant drawback in building and evaluating the prototype. A new plastic composite is adopted for prototypes, Acrylonitrile Butadiene Styrene (ABS) to solve this problem. It is a polymer derived from styrene polymerized together with acrylonitrile in the presence of polybutadiene, and therefore can be defined as a terpolymer. ABS finds application in rapid prototyping machines that use production techniques such as FDM (Fused Deposition Modelling). Because of much lower mechanical properties than the PC, ABS can be used in environments where the temperature is between 0 and $40°C$, more than enough for the scope of a prototype, which is limited to demonstrate a new method and new hardware for SHM.

Rosa Micro S.r.l developed a custom mechanical design following the project requirements. All the components and the electronic PC must be hidden into the support, aiming to protect them from external environmental conditions and aging. Considering that these devices need to work in harsh backgrounds for years, complete protection from water, humidity, sunbeams, and shocks is mandatory. Following the design concept in Fig. 5.2, the mechanic was designed considering two pieces, the magnets holder (*Support 2*) and the electronic support (*Support 1*). The resulting drawing is presented in Fig. 5.5, where supports are connected by a flexible plastic blade placed in the middle of the crack. The holes are at 29 mm from the center and are respectively 58 mm far away from each other. The electronic holder, which is colored in blue, is a waterproof IP67 enclosure holding the Allegro A1319 and its electronics, plus the analog to digital

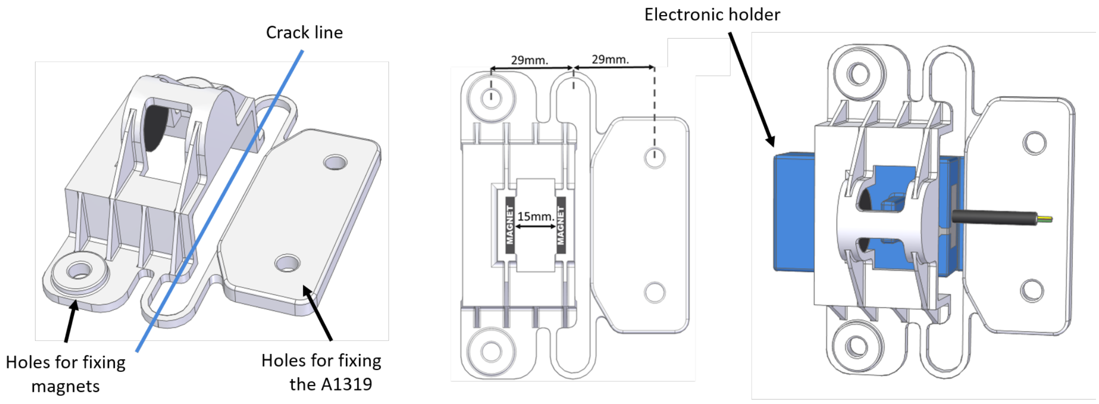


Figure 5.5: Mechanical design of the enclosure. Courtesy of Rosa Micro S.r.l.

converter. Thanks to the plastic blade, the object is easily transportable and installable since it comes in a single piece. However, in response to an applied force, it can perform multi-dimensional movements. The left part of the blue plastic box is stiff, and it is directly leaning to the magnet support, which is rigid and tightly connected with the wall. On the contrary, the right part of the blue plastic box, hidden under the white plastic holes (holes for fixing the A1319) is flexible, allowing an angular movement without affecting the tilting angle between magnets and the Hall effect sensor. This feature is specifically designed to support the issue reported in Section 5.2.1.3, avoiding sensor measurement degradation on tilting movements.

5.4 ADC conversion

Analog to digital conversion, besides requiring a high enough accuracy (i.e., ENOB), needs to minimize measurement dependence on temperature, aging, and noise. To this end, a ratiometric differential approach is implemented in the analog front-end. The schematic uses only a voltage reference (namely REF2033 from Texas Instruments) and a differential sigma-delta ADC embedded on the STM32F373 to minimize the number of components on the board. The circuit is independent of the power supply and stable in temperature with an error of 5 $ppm/°C$ or 8 $\mu V/°C$. The analog to digital conversion is managed by an SDADC, which features a 16-bit resolution and 9 differential analog channels with selectable gains. The conversion speed reaches up to 16.6 ksps for each SDADC when converting multiple channels and up to 50 ksps per SDADC if only one channel conversion is used. The REF2033 is used to power the Allegro A1319 and as a reference voltage for the positive input of the SDADC; the ground is used as a negative reference. Hence, it is straightforward to calculate the SDADC theoretical sensitivity in terms of $\mu V/LSB$. To this end, Eq. 5.4 and Table 5.2 show the STM32F373-SDADC performances.

$$LSB = \frac{V_{ref}}{2^{16}-1} = 50.355\ \mu V. \tag{5.4}$$

The sensor assembly is suitable for measuring cracks in the range of 10 mm ($\pm 5\ mm$), while the Allegro A1319 sensitivity is 14 $\mu m/G$. Hence the theoretical sensitivity is 0.138 $\mu m/LSB$, which degrades to 1.6 $\mu m/LSB$ using the effective ENOB. It is clear that a de-noising procedure must be applied to increase the average sensor performance.

The Allegro A1319 output is calibrated after each power-up, and it is proportional to the supply. Moreover, a static calibration is guaranteed, the low PSRR improves the stability performance under low-frequency dynamic operation, and the spectral noise density (Table 5.2) is below the LSB amplitude (Eq. 5.4). So, this component cannot be considered the primary source of the measurement error. The reference voltage provides an output accuracy of ±0.05% combined with the ratiometric design that makes the sensor output independent from the reference voltage, as shown if Eq. 5.5 and Eq. 5.6.

$$A1319_{out} = (V_{ref+} \cdot \alpha) \text{ where } \alpha = \frac{G}{mV} \in [0,1], \tag{5.5}$$

$$\#\ LSB = \frac{\left((V_{ref+} \cdot \alpha) - \frac{V_{ref+}}{2}\right)}{\frac{V_{ref+}}{2^{16}-1}} = \frac{\left(\alpha - \frac{1}{2}\right) \cdot V_{ref+}}{\frac{V_{ref+}}{2^{16}-1}} = \left(\alpha - \frac{1}{2}\right) \cdot \left(2^{16}-1\right). \tag{5.6}$$

To evaluate analog to digital converter performances, the positive input was connected to an ultra-low noise fixed reference (5 μVpp). The output samples can be represented as a Gaussian probability density function (PDF) with a mean of $-13.8\ LSB$ and variance of 3.3 LSB, generated by a fixed offset and noise signal. The result shows that the ADC errors could be suppressed easily by employing algorithms that need low computational resources, for example the average of the collected data. After verifying the PDF of the freewheel sampling, a series of 12,500 values were evaluated, decreasing the variance to 0.1 LSB, which corresponds to an ENOB equal to 15, increasing the ADC accuracy up to 5.7 times. However, these results are valid only in a controlled environment where the temperature is stable; hence a behavioral study on temperature sweep must be performed to characterize the ADC in all possible conditions. As the crackmeter is a

Table 5.2: STM32F373-SDADC characteristics

Parameter	Value
CLK Frequency	6 MHz
Offset (after calibration)	100 μV
Offset drift	10 $\mu V/°C$
Gain	±2.7%
Integral linearity error	14 LSB
Differential linearity error	1.8 LSB
SNR	92 dB
ENOB	12.5

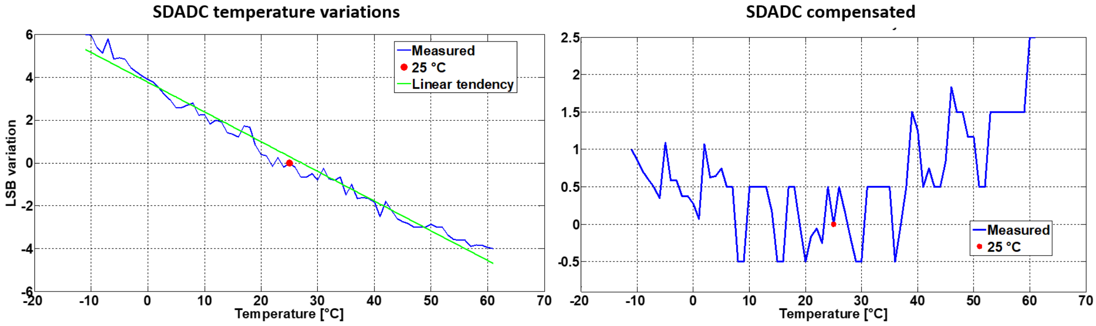


Figure 5.6: SDADC output in raw (left) and compensated (right) mode.

device that has to be exposed to the same environmental condition as regular buildings, its measurement must be insensitive to temperature variations. The most crucial factor affecting thermal stability is the offset error, which is about 10 $\mu V/°C$ for the selected SDADC.

As shown in Fig. 5.6, sweeping the temperature span between $-20°C$ to $60°C$ the LSB variation reaches up to ± 6 LSB, which is equal to an error of ± 0.8 μm. The right plot in Fig. 5.6 makes a comprehensive view of a temperature compensation benefit; indeed, the variation is below ± 0.5 LSB between $-20°C$ to $40°C$, and reaches 2.5 LSB above $50°C$.

This section aims to describe the optimal implementation and the more straightforward way to reach the best performance from the SDADC integrated into the STM32F373. The proposed design respects the project specifications in terms of measurement precision and environmental insensitivity. Moreover, ratiometric architecture should be reliable in long-term life. The analysis demonstrates that, in the static condition, the electronic part of the crackmeter guarantees a sensitivity of ± 0.138 μm at $25°C$ and ± 0.276 μm between $-20°C$ and $-60°C$. The magnetic field variability between magnets is 14 $\mu m/G$, and the A1319 sensitivity is 5 mV/G. This means that an LSB corresponds to 0.138 μm. However, taking into account the integral linearity error, that is not compensated, the sensitivity reported above is valid only around 4000 LSB (552 μm) from the initial calibration value. Outside this confidence range, the maximum error is up to 14 LSB (1.932 μm).

The power consumption of each component was measured, and an aggressive power-saving strategy was implemented. In run mode, with the SDADC working, the MCU at 48 MHz is 29 mA, and the external analog components, together with the Allegro A1319, drain 10 mA. Hence, for a single measurement 129 mJ is required. When it is in sleep, the circuit needs only 1 μW.

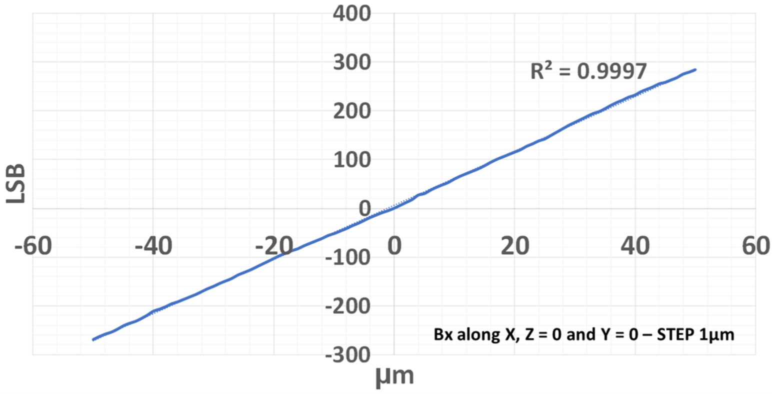
Lastly, it is essential to highlight that these results are extracted in a laboratory with a fixed voltage reference and do not include the A1319 sensor and the plastic enclosure.

5.5 Experimental results

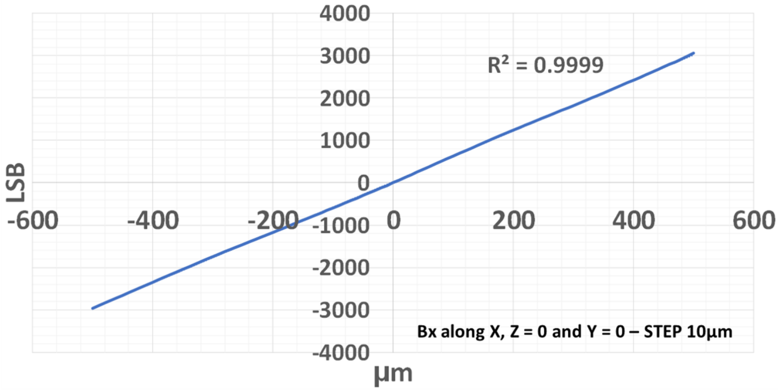
The proposed transducer was tested by specific and stressful assessments to validate the crackmeter behavior under challenging environments and to compare the simulations with real performance. Static checks are performed using a micromanipulator with $< 1\mu m$ resolution, but a thermal chamber was used to simulate harsh and variable working conditions. In the end, an in field evaluation was performed in a civil engineering lab, comparing the crackmeter performances with other commercial products.

5.5.1 Linearity and repeatability

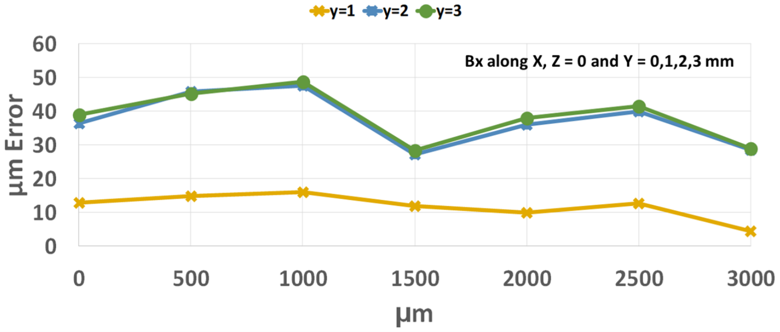
The transducer must be linear in its full scale and measurements need to be repeatable. A series of points were collected to verify the static performance of the crackmeter sweeping the readings between 0 and 3 mm along X, reference at Fig. 5.2. The studied parameters are: linearity in ± 50 μm range at 1 μm step (Fig. 5.7(a)), linearity in ± 500 μm range at 10 μm step (Fig. 5.7(b)), output normalised error varying Y (Fig. 5.7(c)), and repeatability (Fig. 5.7(d)). Results are presented in Fig. 5.7(c) and Table 5.3, where they are compared with simulation results. Linearity deviation is below one micrometer in the ± 50 μm range, while it can reach five micrometers expanding the range. In any case, the MSE (Mean Square Error) is around 1.74 μm, 2× more than what is expected from simulation results; however, the full-scale error varies only of 6‰. The output dependence from Y heavily affects the measurements. In this case, the measured MSE reflects the simulations, which are respectively 29 μm and 26 μm. In 1D crackmeter, this error is not compensated since the orthogonal misalignment check is absent. However, the proposed transducer is a low-cost displacement for long-term deployments and is not designed to be an absolute measuring instrument, but it needs to evaluate the crack variations during his life. As shown in Fig. 5.7(c), the MSE is always positive, so a variation along Y will be read as a crack enlargement, a non-destructive behavior for SHM applications. In any case, the 1D crackmeter cannot be considered an instrument along Y axis since, for 3 mm variation, the output detects only 40 μm (Fig. 5.7(c)). For what concerning the repeatability (Table 5.3), the sensor reaches a maximum of 1.63 μm error and an MSE of 0.60 μm, providing a solid basis to confirm the sensor sensitivity.



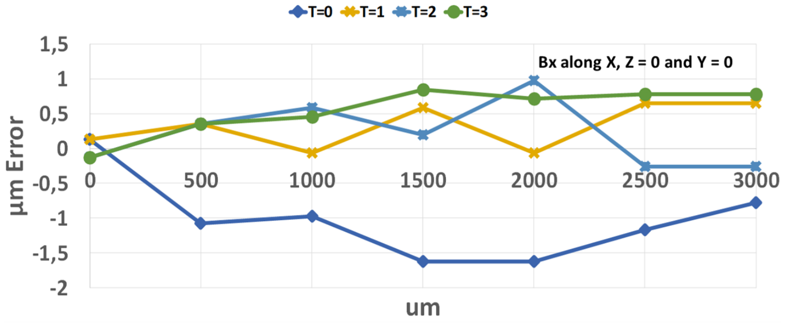
(a) Measured crackmeter linearity in ±50 μm range at 1 μm step.



(b) Measured crackmeter linearity in ±500 μm range at 10 μm step.



(c) Measured output normalised error varying Y, where Y $in0, 1, 2, 3$.



(d) Measured crackmeter repeatability along X, Y and Z euqal to 0.

Figure 5.7: Linearity and repeatability

Table 5.3: Measured crackmeter performance in comparison with simulation results.

		LSB	Gauss [G]	μm	Error % (full scale)
Linearity 50	Max.	6.4	0.06	0.83	0.010
	MSE	2.4	0.02	0.31	0.004
	Sim	-	-	0.2	0.004
Linearity 500	Max.	38.7	0.39	5.03	0.059
	MSE	13.4	0.13	1.74	0.020
	Sim	-	-	0.8	0.014
Var. Y	Max.	376	3.79	48.88	0.815
	MSE	225	2.27	29.28	0.488
	Sim	-	-	26	0.45
Repeatability	Max.	12.5	0.13	1.63	0.027
	MSE	4.6	0.05	0.60	0.01
	Sim	-	-	-	-

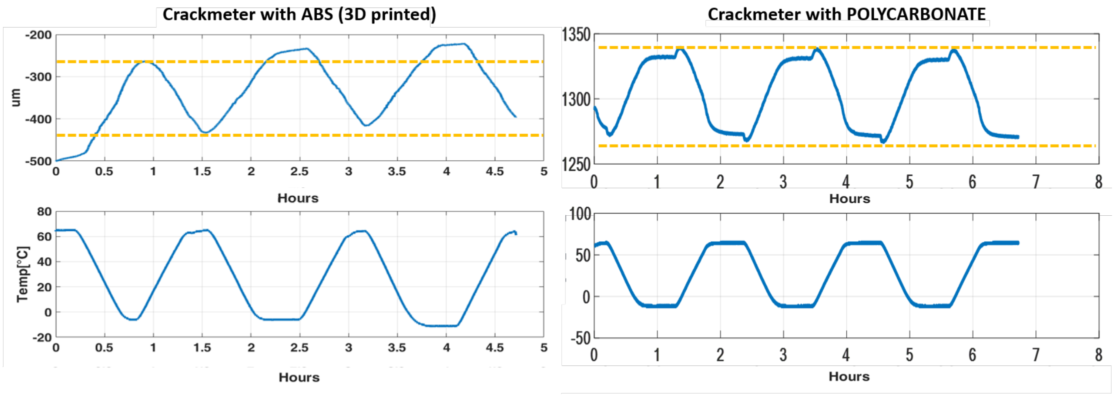
* MSE: Mean Square Error.

5.5.2 Sensibility vs temperature

The plastic support is fixed to a ceramic material, with only 0.37 $\mu m/°C$ of thermal expansion, used as a reference substrate. All the components, included electronics and the enclosure, were measured in a thermal chamber where multiple weather conditions were simulated. The first tests were executed to verify the hysteresis error with significant temperature variation ranges, between -15 to 65$°C$, the maximum range of the thermal chamber. Indeed, these types of issues are difficult to compensate because of unpredictable changes. The primary hysteresis error sources are: (1) Allegro A1319; (2) Non-reversible variations of magnetic field; (3) Non-reversible thermal expansion of PLA box.

The first test was performed by fixing magnets and the Hall sensor directly to ceramic support (without any plastic box or support). The goal was to verify the hysteresis component due to the non-reversible variation of the magnetic field. In full temperature range between -15 to 65$°C$ the sensor behavior is excellent (±5 μm), while in the $7-45°C$ range, the hysteresis component is virtually absent, only ±2 μm.

After that, an evaluation of the overall device with the ABS case was performed. In this case, results show hysteresis, which degrades measurement performances. In the (-10 - 65$°C$) range, an error of ±20 μm is added. In the temperature range of $7-45°C$ the hysteresis error is contained (±5 μm), and the crackmeter features are guaranteed (Fig. 5.8). As previously introduced in section 5.3, the full sensitivity of the device cannot be achieved with the ABS support, which still allows obtaining a very low-cost device. The right plot exhibited in Fig. 5.8 was acquired using polycarbonate support. The


Crackmeter with ABS (3D printed)

Crackmeter with POLYCARBONATE

Figure 5.8: Thermal chamber temperature sweep used to asses the crackmeter temperature dependence.

goal was to verify the hysteresis component derived from the non-reversible variation of polycarbonate. In temperature range between -10 to 65$°C$ the sensor behavior is positively improved compared to ABS support, passing from an error of ±20 μm to ±5 μm. In contrast, in the $7-45°C$ range the hysteresis component is reduced, only ±3 μm. It's possible to observe a peculiar effect when the first derivative of temperature changes; we notice small spikes on the distance measurement, around 5 μm, which introduce a nonlinear error. However, due to the short time window in which it manifests itself, the effect can be considered neglectable since it is reversible.

Applying for linear temperature compensation, the crackmeter output response in long term application, considering Table 5.3 and Fig. 5.8, achieves a sensitivity of ±5 μm or ±30 μm considering millimeter movements along Y in the measurement range of 6 mm and within -20 to 60$°C$.

5.5.3 In-field test and comparison

The crackmeter was assessed in a controlled environment, a civil engineering concrete testing lab. A concrete block with a 50 cm crack was used as a reference structure to compare the crackmeter, an LVDT sensor (Linear Variable Differential Transformer), and the Huggemberg transducer. All the sensors where placed above the crack, as referenced in Fig. 5.9. By applying a force at the concrete block, the crack enlarges, then the output responses are plotted in Fig. 5.9. All the sensors featured the same measurements, and the output curves were overlapped with an MSE error below 5 μm. However, in Fig. 5.9, the responses are not overlapped due to the opening angle of the crack, which was enlarged more on the top, in correspondence with the LVDT sensor, and less on the bottom, where the Huggenberg transducer is placed.

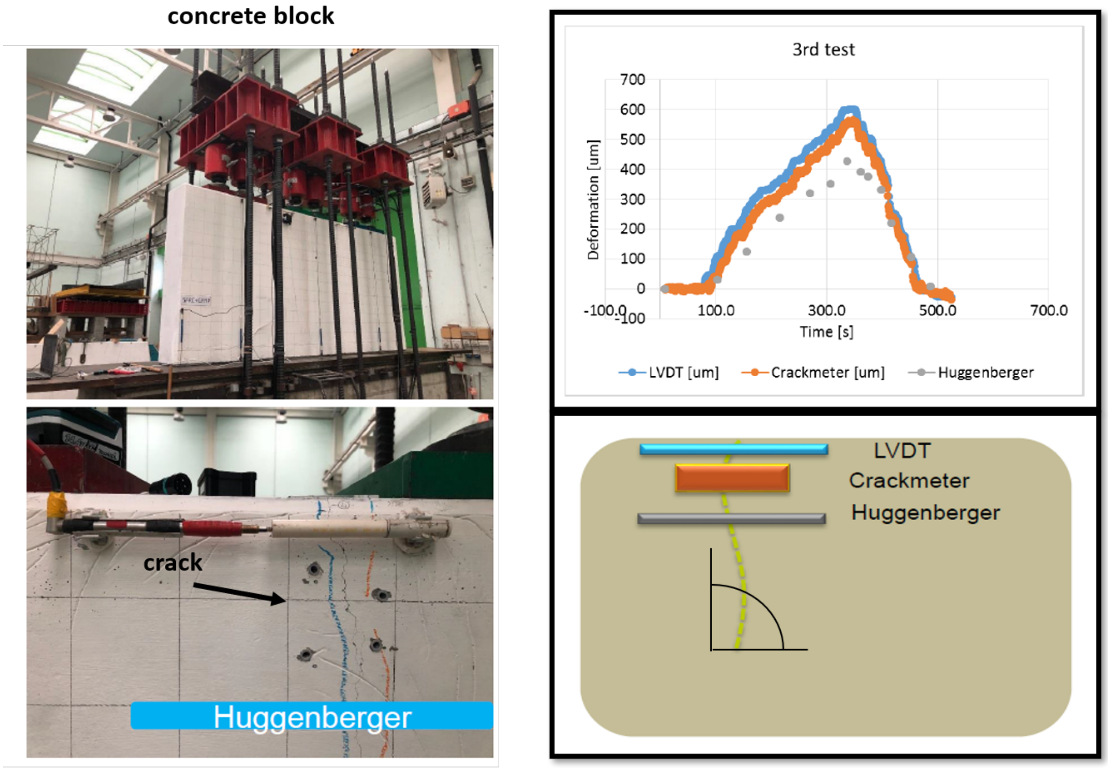


Figure 5.9: Crackmeter assessment and comparison with two commercial products. Courtesy of STMicroelectronics.

5.6 Summary

This Section presented the study and the implementation of a low cost, low power, high precision, magnetoelectric transducer for ASHM applications. In comparison with commercial products, it shows a cost reduction, compared to [167, 169] and [170], or a sensitivity improvement of 10x [166]. The developed prototype can operate under a wide temperature range and can be readily turned into a plug and play product to be used in several SHM contexts. The proposed methodology can be implemented in any topic, not only in SHM. For example, biomedical applications face increasingly complex requirements for human health monitoring, where monitoring vital parameters is a key factor. Hence more and more new bio-transducers are necessary. The Crackmeter is today a commercial product, integrated in an ultra-low power sensor node presented in Section 6.1.

Chapter 6

Low power hardware design for IoT

Remote monitoring and sensing under challenging environmental conditions continues to present problems to prospective practitioners. Lack of infrastructures over which to transmit data and difficulty with the maintenance of battery supplied devices are chief among these problems. Emerging 5G networks, like NB-IoT in Section 3.7, are likely to continue to suffer from connectivity problems in remote areas or in sparsely populated regions where investments are economically unattractive. For this reason, long-range ISM communication protocols are needed, as LoRaWAN that can be deployed everywhere for only the cost of a gateway (Section 3.5.2). Similarly, energy harvesting technologies and advances in low power wireless sensing systems enhance IoT devices durability and reliability, thus reducing the maintenance cost. However, they have yet to deliver on the promise of energy-neutral long-term operation at large scale [168]. This Chapter follows the thesis structure by providing an overview of low power hardware design for IoT sensor nodes. It mainly focuses on such applications that demand stringent working requirements, battery life, coverage, and long term deployments.

Section 6.1 presents an innovative and cost-effective wireless sensor node that tracks cracks in concrete and other construction materials through the transducer described in Chapter 5. The final device exploits LoRaWAN wireless communication and solar energy harvesting, allowing long-term remote monitoring with easy plug and play installation. Experimental results demonstrate that it can achieve about 1 μm accuracy and an expected lifetime of more than 10 years, with stable measurements across a $-15-65°C$ temperature range. Moreover, combining effective power management with a high-efficiency solar energy harvester, it can guarantee more than 10 years of operational life, with 10 measurements per day.

Section 6.2 presents a low-power wireless sensor platform designed to integrate with UAVs to support challenging remote monitoring in hazardous and extreme environments with the total absence of infrastructures. Instead of classical IoT frameworks such as Fig. 2.1, where wired or wireless protocols transfer the data, this network is meant to be physically linked by UAVs, which connect the platform to the cloud. The focus is on low power hardware design, with software developed for two application scenarios in addition to localization and data transfer with robots or autonomous devices. An in-field assessment evaluates the platform capabilities regarding battery lifetime, showing that 30 days of operation between charges is achievable. It is shown that the energy model is valid around 5%, supports adverse weather conditions and provides sufficient energy.

6.1 The low-cost Crackmeter with LoRaWAN communication and energy harvesting capability

This Section presents a cost-effective wireless sensor developed to measure the cracks on multiple buildings, such as private buildings, dams, bridges, or skyscrapers. The transducer is based on the previously described two magnet transducer, characterized by very competitive sensitivity and low-cost components, while the LoRaWAN protocol supports wireless communication. Moreover, by combining efficient power management with a solar energy harvester, the device can reach more than 10 years of operational life, with 10 measurements per day.

The Crackmeter is based on the STM32F373 from STMicroelectronics, which manages the analog to digital conversion and the application layer. Whereas the HopeRF RFM96[1] controls the packet buffering and the LoRa Physical layer. The transceiver provides high interference immunity and ultra-long range spread spectrum communication while minimizing current consumption. Using the LoRa modulation, it can achieve a sensitivity of over -148 dBm combined with the integrated +20 dBm power amplifier, making it optimal for range or robustness applications. The supercapacitor (S-Cap) is run by an L6924D and an SPV1040, capturing power from an external solar panel and ensuring that the S-Cap is not overcharged. The SPV1040 device is specifically designed to efficiently extract power generated from various high output impedance DC and AC without collapsing those sources, generating an average power from μW to mW photovoltaic panels or thermoelectric generators. To keep the current drawn from the battery as low as possible, a smart power management circuit switches between a non-rechargeable battery and the S-Cap. This circuit is usually powered off and is waken up only through synchronous events, like an alarm, and asynchronous interrupts, like

---

[1] https://www.hoperf.com/

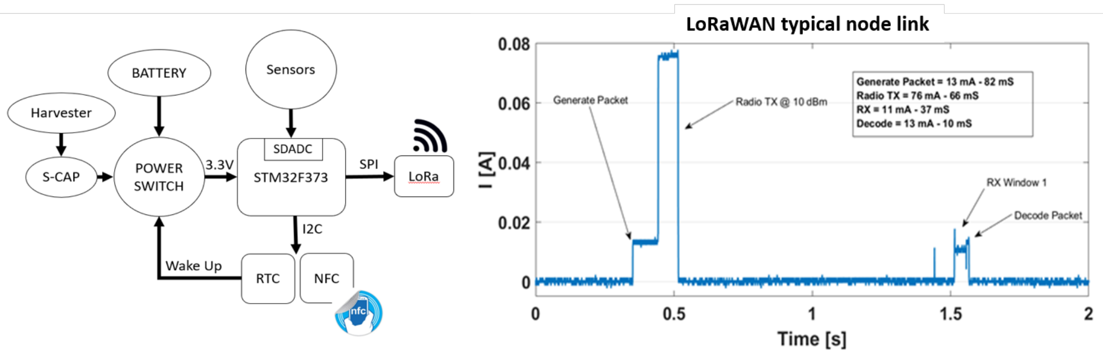


Figure 6.1: Left: Crackmeter block schematic. Right: power consumption of the LoRaWAN uplink.

an external request. A high-efficiency DC/DC converter (TPS61200) generates correct voltages for both analog and digital parts. The TPS61200 provides a low noise power supply from a wide input voltage range, 0.3 to 5.5 V, and an output current of 600 mA. The efficiency at low load is preserved by an automatic power save mode. To save the calibration and the initial configuration, including the LoRaWAN network key, the Crackmeter uses the M24SR64, a dynamic NFC/RFID tag that can be accessed either from the I2C and the RF interface. The M24SR64 controls the interface selection: the RF frontend is based on the ISO/IEC 14443 Type A standard. Moreover, it is compatible with the NFC Forum Type 4 Tag specifications and supports all corresponding commands. A useful open-drain GPO works as a configurable output signal, which is used to trigger the smart power management system by asynchronous events. The GPO pad is enabled when an RF or an I2C session is open; otherwise, it is high impedance. This functionality allows an asynchronous wake-up, used for maintenance or inspections. Finally, an external ultra-low-power real-time clock (M41T82) can wake up the sensor node synchronously, allocating pre-programmed alarms. The M41T82 uses a 32.768 kHz oscillator and includes a watchdog timer, non-volatile time-of-day clock/calendar, two programmable alarms, 8-bit up-counter, and square wave outputs. The eight clock address locations include the century, year, month, date, day, hour, minute, second, and tenths/hundredths of a second in BCD format. In battery mode, it absorbs only 365 nA @ 3 V, 1 μW. An aggressive power-saving strategy was implemented to design an ultra-low power consumption device, and the power consumption of each component was measured. The deep sleep current is 365 nA @ 3 V using the M41T83, which is the only component always powered by the battery. The circuit in sleep mode would use 3.59 μA with the STM32F373 always connected, 10x for the solution adopted. In run mode, with the SDADC enabled, the MCU @ 48 MHz is about 29 mA, and the external analog components, together with the NFC SoC, drain 10 mA. Simultaneously, the RFM96 adds 76 mA in TX (10 dBm output power) to the overall current consumption. The daily energy used is estimated by a precise transient measurement; Fig. 6.1 presents

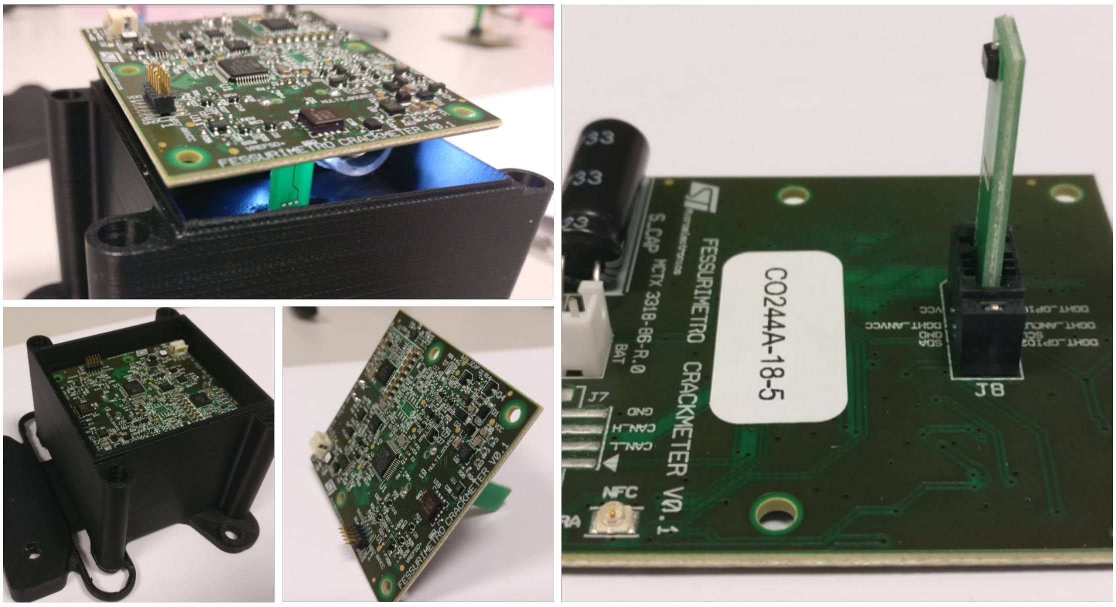


Figure 6.2: The Crackmeter, final version for mass production.

the LoRaWAN power consumption link, where there is a packet transmission followed by one receive window. Moreover, the MCU has to encode the payload, during the data generation, in base64 on top of an AES 128-bit encryption. In the receive window, the MCU confirms the communication link and updates the internal state. Following results in Fig. 6.1, a single supply from a non-rechargeable battery is not enough to guarantee a life over 10 years due to the high current in run mode. The dual-energy source adopted in hardware reduces the battery load, using solar energy to increase the lifespan. The L6924D manages the S-Cap (2 F) in three different manners: (*i*) Pre-charge: when the S-Cap is empty, the charging current is about 10% of the maximum to decrease the energy lost in heat, and consequently, increase the efficiency; (*ii*) Charge: maximum current charge, approximatively 500 mA; (*iii*) End of charge: The voltage reached the maximum allowed. Using a small solar panel (14 cm^2) exposed to direct sunlight of about 32000 lux it is possible to charge the supercapacitor with an average constant current of about 35 mA. With these conditions, the total recharge time is about 234 seconds.

Lastly, in Fig. 6.2, the final version of the crackmeter is displayed, which is ready for mass production and plug-and-play deployment.

6.2 Low power wireless sensor platform for integration with UAVs

This Section presents a low-power wireless sensor platform designed to integrate with UAVs, aiming to support challenging remote monitoring in hazardous and extreme environments with the total absence of infrastructures. Instead of classical IoT frameworks such as Fig. 2.1, where wired or wireless protocols transfer the data, this design is meant to be physically linked by UAVs, which connect the platform to the cloud, see Fig. 6.3. The focus is on low power hardware design and software support for smart supply management in conjunction with localization (Section 4.1). An in-field assessment of the platform capabilities regarding battery lifetime is provided, showing that 30 days of operation between charges is achievable. Section 6.4.3 shows that the energy model is valid around 5% and supports adverse weather conditions providing sufficient energy, even where an issue occurs. As interoperability and flexibility are at the core of scientific research, the whole project is released as a fully open-source project on GitHub[2].

Recent advances integrating unmanned aerial vehicle (UAV), or *drone*, platforms with *in situ* battery supplied sensors can enable long-lasting monitoring systems in remote and extreme environments [171], where the former is responsible for both data collection and energy replenishment of the latter. Under such conditions, it is assumed that the UAV may be supported from a recharging station with an internet connection to service a sensing field that may span many square kilometers. This poses several relatively novel challenges, including the design of appropriate communication and localization mechanisms. Indeed, remote sensing devices may be tasked with collecting a variety of heterogeneous data sets to be retrieved by an UAV at periodic intervals while at the same time recharging the battery or a super-capacitor.

In this scenario, communication protocols should also contribute to precise localization in support of wireless charging, in addition to low power operation and high throughput [152, 172], where, for example, a sub-meter accuracy is required for inductive wireless power transfer (WPT). Although common transceivers implementing the aforementioned wireless technologies offer a measure of received signal strength indicators (RSSI) that can be used for ranging. However, it is well understood that this measure is insufficiently accurate or granular for the purposes of the proposed scenario, where a sub-meter localization is mandatory [173].

The application specifications required in this section are twofold, precision support for UAVs and fast bi-directional data transfer, enabling WPT for remotely deployed devices. Qin *et al.* provides a pratical basis to develop efficient wireless power transfer

---

[2] https://github.com/tommasopolonelli/SynthSense-WSN-UAV

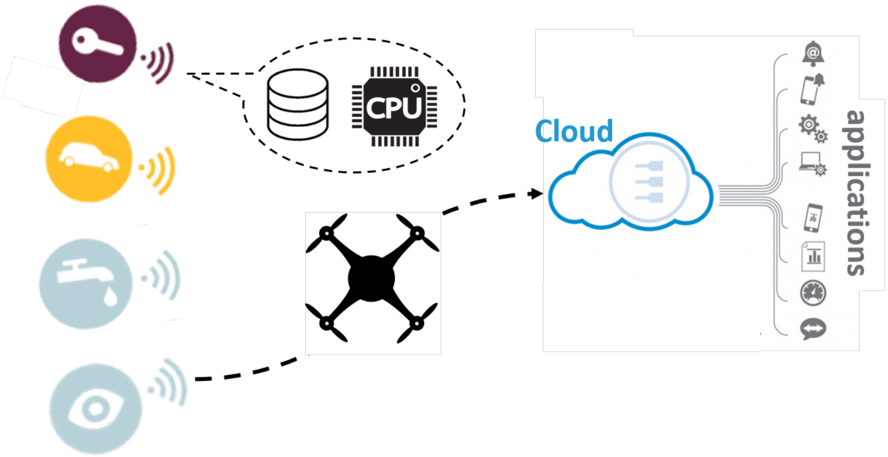


Figure 6.3: Top-level overview of a IoT sensor node supporting UAVs.

mechanisms using a combination of GPS, Bluetooth and WPT to find the optimal landing position from which to transmit power, proposing the first concept of a three-stage localization mechanism in [152]. Due to the adopted RSSI method for the state estimation, they suffer from a high packet loss, in reality relying only on the GNSS [152].

Surprisingly, there are no *off-the-shelf* wireless sensor devices that incorporate a UWB wireless interface. Therefore, it was necessary to design a *first-of-its-kind* hardware and software platform, provided open source[3] to the community, that is flexible to meet the needs of a variety of application scenarios and to enable UAV-sensor data collection and power transfer. The objective was to leverage the advantages of existing communication standards with low network maintenance costs in combination with emerging UWB technology, offering further advantages in innovative and challenging WSN deployments, handling lower per-bit energy, and high precision in ranging performance. Moreover, the platform combines elements of the relative localization system based on the UWB' ToF (Time of Flight) evaluation [174] with results from [175, 176], which describe a high efficiency (70%) inductive power transfer system (IPT), to create a system capable of 150 W instantaneous WPT (Wireless Power Transfer). Structural health monitoring or other event detecting applications need more than hundreds of Hz, whereas soil moisture monitoring or corrosion detection employment, for example, may require only a few readings per day. Therefore, the platform has been designed to provide the capabilities of collecting, processing, storing, and transmitting sampled data from a variety of sensors. Thus, this board is meant to facilitate a variety of sensing tasks with many different sampling frequencies, in conjunction with a throughput that must be sufficient to deliver the data to the UAV within the charging period.

---

[3] https://github.com/tommasopolonelli/SynthSense-WSN-UAV

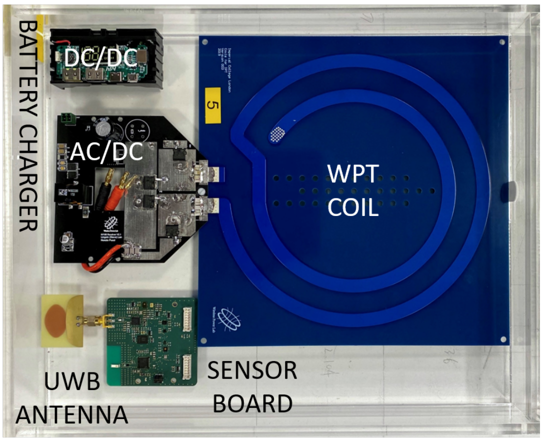


Figure 6.4: Complete receiver side hardware. The coil used for wireless power transfer (blue), sensor board (green, bottom left) and battery management circuits (DC/DC and AC/DC; black boxes) are shown. The UWB antenna is directly connected to the sensor board.

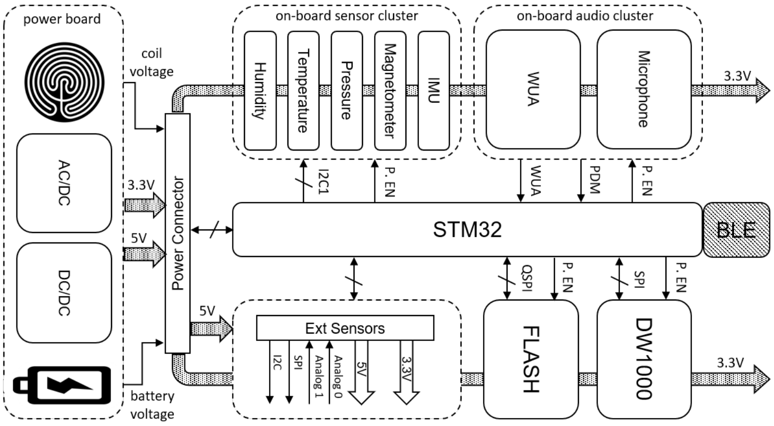


Figure 6.5: Sensor board schematic. The power connector splits the power electronics and sensor boards.

6.3 Hardware

6.3.1 Technical specification

The hardware platform comprises two PCBs, the sensor board and power electronics board (PEB), respectively; shown in Fig. 6.4. The two boards are connected via a 16-pin JST connector, shown schematically in Fig. 6.5 and pictured in Fig. 6.4. The power electronics board deals with tasks related to the power supply, including generating reliable and stable 3.3 V and 5 V DC voltage from two Panasonic 18650 batteries, managing fast-charging and the wireless power transfer system that supports up to 150 W.

The PEB manages the wireless power delivered by the charger [175, 176]. These papers include operating principle and performance characterization. Under the proposed scenario, the WPT system consists of two coils; a receiving coil incorporated into the PEB and a transmitting coil connected to the UAV. The magnetic field is generated by driving the transmitting coil with an alternating current, which induces a voltage across the PEB and is coupled with the receiver coil. The transmitting and receiving coils have different sizes and are separated by variable distance and different misalignment degrees. The coefficient k describes the coupling between the two coils, which is variable in time and distance during the UAV approach:

$$k = \frac{M}{\sqrt{L_p L_s}}. \tag{6.1}$$

the mutual inductance is defined as M, which decreases when increasing the distance between two coils, i.e., the distance between the drone and sensor node, while L_s and L_p are respectively defined as self-inductances of the receiving and transmitting coils. The accurate expression of M is described in [177] together with its numerical calculation. Monitoring the receiver inductive voltage is helpful to align both coils correctly; indeed, a higher value of k (up to a maximum of 1) indicates that a large portion of the magnetic field generated by the TX coil is coupled into the sensor node coil. Hence the sensor board has a dedicated analog channel connected to the receiver coil, as shown in Fig. 6.5. Consequently, the UAV makes use of a look-up table that correlates distance with the measured voltage to estimate the coil-to-coil misalignment, similar to [178]. This method needs an initial calibration but enables a real-time reliable distance estimation. As the details of the PEB are available in [175] and [176], the remainder of this section focuses on the sensing and wireless aspects of the sensor node, which facilitates precise localization to ensure high-efficiency coupling, high data throughput, and energy-efficient operation.

6.3.2 Sensor Node

The proposed design features many complementary sensors, internal and external, and a multi-protocol radio front-end, to support various sensor-fusion and UAV applications. All the components are fitted into a 70 x 70 mm PCB, combined with the external 30 x 40 mm UWB antenna. The sensor node needs only low-cost off-the-shelf components, a prerequisite for open source projects. Moreover, it is specifically designed to be a general-purpose and versatile device, in conjunction with unique optimization to be used in combination with drones. However, it can be easily deployed in other application scenarios, such as Industrial Internet of Things and smart agriculture.

The sensor node supports UWB, compliant with IEEE 802.15.4-2011, for 2-way ranging and data transfer at a rate up to 6.8 Mbps in addition to Bluetooth 5.0. An integrated STM32WB55RGV (STM32 hereafter) from STMicroelectronics manages all the stacks and sensors. This MCU has the advantage of low power consumption, as well as excellent peripherals support. The ARM Cortex-M4 is used for the main processing tasks, while the ARM Cortex-M0 is the radio communication protocols engine. A VFQFPN-68 packaging was chosen for ease of soldering and GPIO expansion capabilities. To decrease the power consumption in sleep and deep-sleep states, the circuit implements four voltage supply domains. The STM32 enables or disables each domain depending on application requirements and operating modes. A TPS27081A manages the switching, supporting a maximum continuous current of 3 A. The board integrates one temperature (TMP117) and one humidity (HDC2080) sensors that provide high accuracy measurements, respectively 0.2°C and ±2%, with meagre power consumption, 1.65 μW on average. A digital pressure sensor from Bosch (BMP280) with an average current consumption of 2.7 μA at 3.3 V (at 1 Hz sampling rate) is included. To expand application support, the board integrates two high-frequency sensors, a 3D inertial IC (LSM6DSOX) and a digital magnetometer (IIS2MDC) from STMicroelectronics. The LSM6DSOX is a system-in-package device featuring event-detection hardware that enables efficient motion tracking and contextual awareness, implementing automatic recognition of free-fall events, click and double-click sensing, activity or inactivity detection and pedometer functions. The combined current of IIS2MDC and LSM6DSOX is 600 μA at 3.3 V. The sensors mentioned above are in INT_VDD voltage domain[4] and use a shared I2C bus at 400 kHz. In addition to the onboard ICs, the device can incorporate external analog and digital sensors through a 16-pin connector (*Ext Sensors* in Fig. 6.5), for example, a pH sensor (SEN01169), dissolved oxygen sensor (SEN0237-A), and a 5M pixel CMOS camera (MT9P401) with serial SPI interface.

In addition to the aforementioned environmental sensors, the sensor board enables audio recording. A digital microphone (IMP34DT05) from STMicroelectronics combined with an ultra-low-power wake-up audio (WUA) sub-circuit (capable of waking up the MCU on triggering events when the environment sound is above a predetermined threshold) are included. Exploiting the wake-up method, the MCU can store only relevant information and remain in sleep mode during "inactivity" periods. The IMP34DT05 has an acoustic overload point of 122 dB SPL with a 64 dB signal-to-noise (SNR) ratio and a sensitivity of -26 dB FS at 94 dB SPL, 1 kHz. It is well-known that sound management is a challenging requirement in low power devices. Indeed, it needs continuous and high-frequency sampling that drains a non-negligible amount of current from the energy buffer. The MCU and non-volatile memory must continuously work to elaborate, compress, and store the microphone data stream. On the other hand, a

---

[4] `https://github.com/tommasopolonelli/SynthSense-WSN-UAV/blob/master/SensorNode/HW/Schematics/SensorNode.pdf`

duty-cycled approach is not feasible since it is likely to generate low-quality records or miss events entirely. The MP23ABS1 is a compact, low-power analog microphone built with a capacitive sensing element, and it is the main WUA component together with a micro-watt comparator (LTC6259). It has an acoustic overload point of 130 dB SPL with a typical 64 dB signal-to-noise ratio. Its sensitivity is -38 dBV at 94 dB SPL, 1 kHz. The amplitude of the analog output signal depends on acoustic pressure. The relationship between sensitivity (S_o) and gain (G_{micro}) at 1 kHz is given in the MP23AB02B datasheet and is calculated using Eq. 6.2, where S_o is measured in dBV/Pa and G_{micro} in V/Pa. In terms of scale, 20 μPa is the minimum 'hearing' level, while talking produces an acoustic pressure around 20 mPa.

$$G_{micro} = 10^{\frac{S_o}{20}} \tag{6.2}$$

Assuming free field conditions, if we double the source distance, the acoustic pressure is halved if the source is considered a point like a speaker; or $\sqrt{2}$ if the source is considered to be a line such as a road.

The maximum gain setting G_{max} is calculated using Eq. 6.3 by considering the maximum sensitivity of the microphone S_{o_max}, the maximum output root-mean-square amplitude needed after the amplification stage V_{rms_max}, and the maximum acoustic pressure P_{max} to be detected. In this design, P_{max} is equal to 94 dB SPL or 1 Pa. Since the power supply voltage is 3.3 V in the audio power domain, V_{p_max} must be less than 1.65 V.

$$\begin{cases} V_{rms\_max} = 1V \\ V_{p\_max} = V_{rms\_max} \cdot \sqrt{2} = 1.4142 \\ G_{max} = \frac{V_{rms\_max}}{10^{\frac{S_{o\_max}}{20}} \cdot P_{max}} = 112 \end{cases} \tag{6.3}$$

Considering a 10 kHz waveform with 3 V_{pp} amplitude, the amplifier minimum slew rate must be 0.19 V/μs. The LTC6259 is a dual operational amplifier with low noise, low power, low supply voltage, and rail-to-rail inputs and outputs. It is unity-gain stable with or without capacitive loads. It has a gain-bandwidth product of 1.3 MHz and 0.24 V/μs slew rate while drawing 20 μA per amplifier. With a 1.3 MHz GBP and BW of 10 kHz, the maximum gain is 130, which is greater than G_{max}. The raw signal is amplified by G_{max} and filtered with a first-order 10 kHz RC passive filter. Finally, a comparator detects if the audio level is above the threshold, which is set to 60 dB SPL in our experiments. The WUA requires only 160 μA at 3.3 V, drastically reducing the average power consumption of the audio recording and the size of non-volatile memory required (more details in Table 6.2). The WUA performances are shown in Fig. 6.6, where the WUA output and the digital microphone are compared to measure the circuit false negatives, which are negligible above 70 dB SPL.

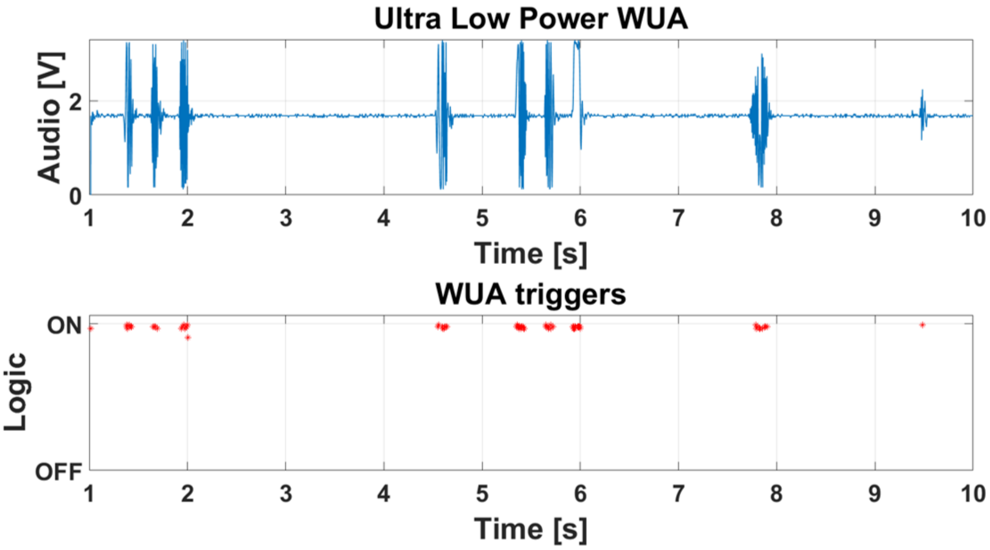


Figure 6.6: WUA operations compared with the digital microphone. Red dots mean the wake up trigger for the STM32, which instantly records the audio. In this graph, up to 80% of the recording energy is saved due to the WUA capabilities.

The MCU internal flash is mainly used to store the firmware, and it is not sufficient for data-intensive continuous sampling storage, for example, audio recording at 16 ksps. Thus, an external flash (MT25QL256ABA) of 256 MB is used to store sensor data.

A detailed power consumption analysis and its corresponding model are presented in the following section.

6.4 Energy model

Energy consumption and data accumulation profiles for the sensor node are modeled in this Section to assist in scheduling operations for data collection and power transfer. The sensors, coupled with their working principles, and given different application scenarios, result in a wide variation in average energy consumption. Aggressively duty-cycled sensors consume negligible energy compared with other sensor node components, such as temperature and humidity, commonly sampled at sub-Hz frequency. On the contrary, sensors with high sampling frequencies draw more power and result in intensive data generation that contributes to the complexity of the working pattern. Hence this section proposes a hierarchical method, which considers both perspectives of energy and data. Data collection and wireless power transfer could be scheduled accordingly. The sensor node energy model comprises three subsections. RF energy modeling deals with energy consumption only happens within the RF core, such as transmitting and receiving a packet. CPU energy modeling deals with the CPU energy consumption. Sensors modeling considers the energy profile of all sensors on board.

6.4.1 System CPU energy modelling

The system CPU features three operation modes: active, idle, and standby. The active mode consumes the most current, where all functionalities are enabled, while the standby state utilizes the least, but no operation is allowed. In idle, the core stops, but peripherals and ADC are still active. Table 6.1 denotes the I_m^{MCU} current, where m represents a working mode $m \in \{active, idle, stop\}$. In addiction to I_m^{MCU}, the transition current among modes is denoted as I_t^{MCU}. The energy absorbed during MCU operation can thus be written:

$$E_{MCU} = \sum_m V_s I_m^{MCU} T_m^{MCU} + \sum_t V_s I_t^{MCU} t_t^{MCU} N_t^{MCU} + V_s I_{BLE}^{MCU} T_{BLE}^{MCU}. \tag{6.4}$$

where T_m^{MCU} is the sum of operation time of a mode. The fixed transition time between modes is denoted by t_t^{MCU}. N_t^{MCU} is the number of transitions carried out during a period. V_s is the supply voltage. I_{BLE} and T_{BLE}^{MCU} take into account the BLE transceiver and its dedicated ARM Cortex-M0.

6.4.2 UWB subsystem energy modelling

A probabilistic approach to modeling a device incoming and outgoing information flow can be shifted to a deterministic effort. It can be expected that the radio channel is silent for a long time, becoming busy only on UAV arrival and synchronization. The UWB RF transceiver then shifts its modes between active and sleep. Transmitting and receiving currents are denoted $I_{p,tx}$ and $I_{p,rx}$ respectively, where the subscript p indicates the radio configuration. The transition current from tx to rx, or rx to tx is defined as I_{swh}.

It is assumed the data volume to be sent to the UAV for a specific sensor is PAY_{tot} in bytes, where PAY_{max} is the maximum number of bytes per transmission. Given a data rate r_p, where p represents a specific $Mode(p)$ data rate, such as r_2 in Table 6.1, the time consumed for transmission is simply the division of the two. The acknowledgement time t^{ack}, ranging beaconing time t_p^{rang}, and tx-rx switching time t^{swh} are configuration dependent (cf. subscript p). It is assumed that N^{ack} acknowledgments are received during data transmission only if A_s is equal to 1, otherwise null. The RF transmission energy consumption can then be modelled as in Eq. 6.7, where $E_{Ranging}$ in Eq. 6.5 and E_{Data} in Eq. 6.6 are respectively the overall energy used for the two-way ranging protocol and the sensor node data transfer. In Eq. 6.5, a duty-cycled listen mode designed to

Table 6.1: STM32 and DW1000 details

Symbol	Description	Value
V_s	sensor board voltage supply	3.3 V
I_{active}^{MCU}	STM32 run current	7.59 mA
I_{idle}^{MCU}	STM32 idle current	4.15 mA
I_{stop}^{MCU}	STM32 stop current	2.45 μA
I_{t}^{MCU}	STM32 transition current $stop \rightarrow run$	7.59 mA
t_{t}^{MCU}	STM32 transition time $stop \rightarrow run$	5.71 μs
I_{BLE}^{MCU}	STM32 BLE current advertising (Tx = 0 dBm; Period 1 s; 31 B)	30 μA
$Mode(p)$	DW1000 Op. Mode	$p \in \mathbb{N}, p \in [2,3]$
I_{swh}	DW1000 transition current	4 mA
I_{sleep}	DW1000 sleep current	1 μA
t_{swh}	DW1000 transition time	500 μs
t_{ack}	DW1000 acknowledgement window	50 ms
BW	DW1000 bandwidth	499.2 MHz
$I_{2,rx}$	DW1000 mode 2 receive current	244 mA
$I_{2,tx}$	DW1000 mode 2 transmission current	69 mA
EPB_2	DW1000 mode 2 EPB	44 nJ
r_2	DW1000 mode 2 datarate	6.8 Mbps
PRF_2	DW1000 mode 2 pulse repetition frequencies	16
$Preamble_2$	DW1000 mode 2 preamble size	128
$Code_2$	DW1000 mode 2 preamble code	3
CH_2	DW1000 mode 2 frequency	3993.6 MHz
$I_{3,rx}$	DW1000 mode 3 receive current	191 mA
$I_{3,tx}$	DW1000 mode 3 transmission current	50 mA
EPB_3	DW1000 mode 3 EPB	2.1μJ
r_3	DW1000 mode 3 datarate	110 kbps
PRF_3	DW1000 mode 3 pulse repetition frequencies	64
$Preamble_3$	DW1000 mode 3 preamble size	1024
$Code_3$	DW1000 mode 3 preamble code	9
CH_3	DW1000 mode 3 frequency	4492.8 MHz

Table 6.2: Sensors energy modelling

ID	Sensor	Model	Max Op. Frequency	T_{active}	EPS
a	Light	OPT3007	10 Hz	1.9 ms	56 μJ
	Temperature	TMP117	1 Hz		
	Humidity	HDC2080	1 Hz		
	Pressure	BMP280	1 Hz		
b	Accelerometer	LSM6DSOX	9 kHz	7.0 ms	265 μJ
	Magnetometer	IIS2MDC	100 Hz		
c	Microphone	IMP34DT05	2.4 MHz	cont	26 mJ (16 ksps)
d	Microphone	MP23ABS1	20 kHz	cont	99 μJ (1 s)
e	Coil Voltage	STM32 - ADC	10 kHz	cont	1 mJ (1 s)

reduce the average power consumption is considered, the DC variable spans between 1% to 7% for the application scenario, and t^{DC} indicates the on time.

$$E_{Ranging} = V_s I_{p,rx} t_p^{rang} + V_s I_{p,rx} t^{DC} DC, \tag{6.5}$$

$$E_{Data} = 8 \cdot PAY_{tot} EPB_p + A_s ceil\left(\frac{PAY_{tot}}{PAY_{max}}\right) \cdot \left(V_s I_{swh} t_p^{swh} + V_s I_{p,rx} t_{p,ack}\right), \tag{6.6}$$

$$E_{rf} = E_{Data} + E_{Ranging}, \tag{6.7}$$

$$t_p^{rang} = \frac{d'}{v} - \frac{\kappa}{d'} ln\left(\frac{0.01}{d'}\right). \tag{6.8}$$

Eq. 6.5 considers the ranging settings used by the UAV by approaching the sensor node. However, as it is dependent on vehicle speed and update rate, both of which vary in real-time due to weather conditions and drone control algorithm. In Eq. 6.8 an ideal case scenario with exponential deceleration between κ and the sensor node is modeled, considering hitting the first receive windows at d' distance. t_p^{rang} is defined as the time required for the vehicle to reach the sensor from the initial starting point.

6.4.3 Sensors Energy Modelling

This section provides the energy consumption model of the internal sensors, expressed as energy per sample (EPS), a value that is independent of the acquisition frequency. Low-frequency sensors, such as light, temperature, humidity, and pressure, are considered as a single block, which needs 56 μJ and 1.9 ms to collect the measured environmental conditions; see Table 6.2. Inertial measurements are from LSM6DSOX and IIS2MDC, which need 7 ms and 265 μJ to obtain 9-axis, see Table 6.2 for further details.

The audio sub-block energy consumption is given as EPS, defining one sample as one second of operation, yielding ∼16k samples. DC_A is defined as the expected audio duty cycle, which is application dependent. It describes the percentage of operation in which the sound level is above the WUA threshold. Eq. 6.9 shows the sensor energy profile, excluding the external sensors that may be connected to the device and the WPT coil voltage (Table 6.2 - e). In Eq. 6.9, N_s^x is the number of samples per application reference period.

$$E_{Sensors} = EPS^a N^a + EPS^b N^b + EPS^c DC_A + EPS^d (1 - DC_A). \tag{6.9}$$

Finally, the storage energy is modeled by the sum of the time required to erase a sector and the time required to write the data. The speed is assumed to be b_a, where $a \in \{read, write, erase\}$. Table 6.3 shows the equivalent EPB with a standard page-size (256 bytes). As expected, the erase power consumption heavily increases the average EPB, hence it is more convenient to clear the whole memory while the sensor node is powered by the mains or is under charging by WPT.

Table 6.3: Flash energy modelling

Device	Model	Operation	b_a	EPB
Ext. Flash	MT25QL256ABA	$write + erase$	80 kB/s	673 nJ
Ext. Flash	MT25QL256ABA	$write$	2 MB/s	1.61 nJ

6.5 Experimental results

Together with the hardware project and the source code, two application examples are provided[5]. To validate the energy model, battery lifetime estimation of the APP1[6] is presented, which consists of sampling and storing the internal sensor ID a and b (Table 6.2 and Table 6.3) at fixed frequency, 1 Hz and 12 Hz respectively. In this application, the sensor node expects to be recharged at programmed intervals, to restore the battery level and to collect the measured data. The DW1000 DC is 1%, and the autonomous vehicle approaching speed v is programmed at 2 m/s. Uplink acknowledgements are disabled, hence A_s is null. The STM32 does not apply any algorithm on the collected data, so it returns in *stop* mode between each sample. Lastly, the audio recording is turned off while the WUA is always active. Lastly, t_p^{rang}, correspondent to ATWR time

---

[5] `https://github.com/tommasopolonelli/SynthSense-WSN-UAV`

[6] `https://github.com/tommasopolonelli/SynthSense-WSN-UAV/tree/master/SensorNode/SW/APP1_IntSensor`

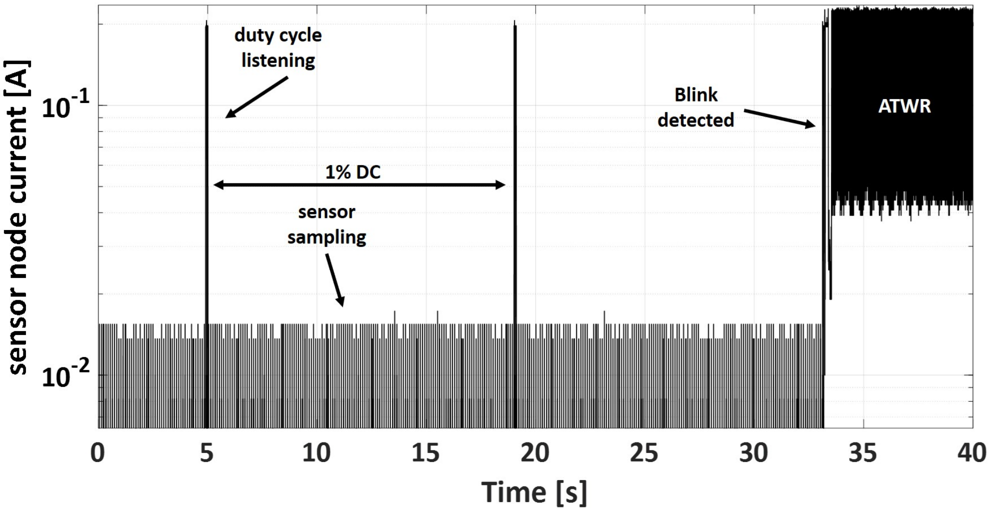


Figure 6.7: APP1 current profile during: sensor sampling, DC listening, and ATWR mode.

in Eq. 6.8, is equal to 26 s. Using these configurations, the average energy calculated by Eq. 6.10 is 79 J for one hour of operation, t^{DC} equal to 3600 s. In one hour, sensor board collects more than 5 MB, using $(E_{MCU} + E_{Sensors})$ approximately 9.7 J, whereas the DW1000 in duty cycle mode needs 22.2 J.

$$E_{tot} = E_{MCU} + E_{rf} + E_{Sensors} \tag{6.10}$$

$$Days = \frac{E_{Battery}}{E_{tot}}. \tag{6.11}$$

With the proposed application settings, the estimated battery lifetime using Eq. 6.11 is approximately equal to one month.

The proposed model was verified assessing the sensor node behaviour in laboratory (static and controlled conditions) and in real environments. The measured average current consumption regarding the internal sensors acquisition is 0.84 mA, equivalent to 9.98 J, while the DW1000 needs 1.95 mA for 1% DC listening, which corresponds to 23.2 J. Compared with the presented model, averaging on one hour of operation, these values differs by 3% and 4% respectively. These values are reflected in Fig. 6.7, where the logarithmic plot of the current profile shows all sensor node functionalities. Continuous sampling generates the 10 mA spikes, while the 190 mA spike comes from the DC listening of the DW1000. Finally, after a correct reception of a Blink message from the UAV, the ATWR is enabled, alternating transmission (lower current) and reception (higher current). Data transfer is not shown in Fig. 6.7, but is equal to a constant transmission current of $Mode(3)$. Fig. 6.8 presents a drone landing example, showing that it can land only 38 s after the first Blink message was received by the sensor node (point zero in ATWR and WPT Distance). Fig. 6.8 shows the same test from two

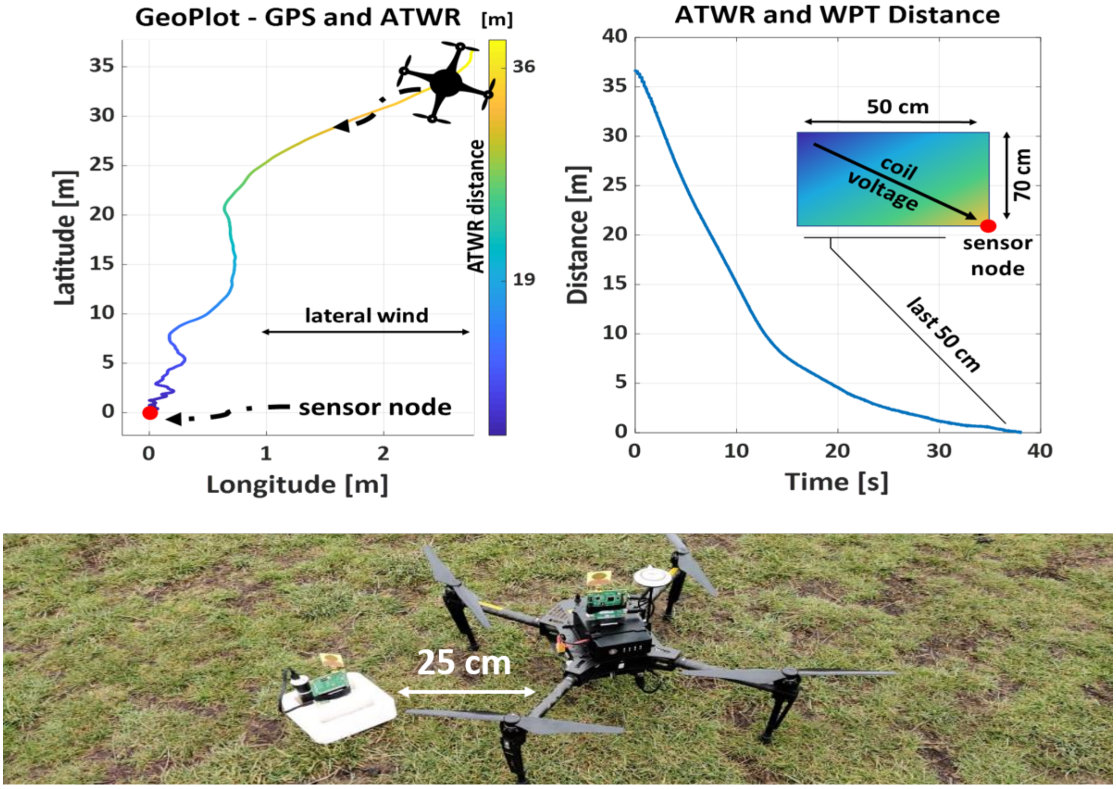


Figure 6.8: Experimental results from in-field tests of the hardware platform. Left: drone path in the presence of lateral wind, where ATWR defines the colorbar. Right: Estimated point-to-point distance using ATWR. Bottom: The drone correctly lands using the UWB support (not to scale).

perspectives; on the right, distance estimation using both ATWR and WPT is presented, while on the left is plotted the planar route using the GPS and ATWR, which define the colorbar. These values are acquired using the drone inertial module and its internal storage (Fig. 6.5). In this specific deployment, the environment was characterized by a strong lateral wind. Its speed reached up to 5 m/s. In this case, $E_{Ranging}$ becomes 24 J while the estimation is 45 J. In this application scenario, overestimating the landing energy cannot be considered an error. Indeed, in a real environment, the UAV needs to change direction many times to correct the effects of the wind lateral force, consequently extending the approach time. Due to environmental wind instability, it is recommended to keep an energy buffer if the drone needs more time to land correctly. Indeed, if the battery is completely discharged, the drone will be unable to land close to the IPT coil, since the UWB will be unreachable. This effect is further increased during the final meters of the approach, where the speed is reduced and the propellers rotate more slowly, making it stronger challenging to combat the wind.

Finally, a landing precision of 25 cm was achieved even with a strong lateral wind. A reliable estimation of the battery lifetime, in conjunction with the memory use, is a fundamental factor for the entire system. Indeed, the drone must visit each sensor within a maximum period given by Eq. 6.10 and Eq. 6.11.

6.6 Summary

This Chapter presented two low-power wireless sensor platforms designed with particular applicability in hazardous and extreme environments, where infrastructure communications may not exist, and devices maintenance is difficult for human operators. The SHM sensor node makes use of the LoRaWAN protocol, studied and enhanced in Chapter 3, and the custom transducer presented in Chapter 5. Experimental results demonstrate that it achieves micro-meter accuracy and a lifetime of more than 10 years, with stable measurements across a $-15 - 65°C$ range. Moreover, it can guarantee more than 10 years of operational life in harsh environments, with 10 measurements per day and a wireless coverage above 10 km. The UAV platform features localization systems that exploit ATWR to improve position estimation to sub-meter precision, using a single reference point. The focus is on hardware design, with software developed for two application scenarios in addition to the landing system. Modeling and in-field assessments of the platform capabilities regarding battery lifetime are provided, showing that 30 days of operation between charges is achievable for a sample application. It is shown that the energy model is valid around an error of 5%, supports adverse weather conditions and provides sufficient energy even where a drone needs to adjust the landing path multiple times. This provides a framework supporting scheduling the UAV visits, which must occur before the energy buffer is depleted.

Chapter 7

Edge computing

The increase of IoT devices at the edge of the network produces a massive amount of data to be sent, computed and stored, pushing network bandwidth requirements to the limit [60]. Despite the improvements in network protocols, data centers struggle to guarantee acceptable throughput and response time, which is a critical requirement for many applications. On the other hand, devices at the edge continuously consume data from the cloud, forcing companies to build content delivery networks to decentralize storage and service provisioning, leveraging physical proximity to the end-user. This dissertation tackles one of the most critical task, collection and streaming of images or video. In the following sections, two case studies show image processing on the edge using a low-power RISC-V MCU. The first introduce the topic of JPEG compression to decrease the bandwidth and the power consumption. The latter makes use of an entire image processing pipeline to extract relevant features directly on the edge, using quantized neural networks and edge-optimized algorithms.

7.1 An Energy Optimized JPEG Encoder for Parallel Ultra-Low-Power Processing-Platforms

The energy autonomy and the lifetime of battery-operated sensors are primary concerns in industrial, healthcare, and IoT applications, in particular when a high amount of data needs to be sent wirelessly. In this scenario, Wireless Camera Sensors are usually left in the environment to acquire and transmit visual data [179, 180]. From a system-level viewpoint, the energy consumption is dominated by the radio subsystem and is proportional to the number of bytes to transfer [27, 29, 181]. Concerning WCSs, on-board real-time image compression is the appropriate solution to decrease the system energy [51, 182]. In fact, bringing the intelligence close to the sensor enables the reduction of transmission cost thanks to the compression of the data dimensionality [183].

Executing computationally heavy tasks, such as an image compression pipeline, without assuming a dedicated hardware acceleration engine (which may not be available or affordable for cost reasons) typically requires adequate computing capabilities and a large memory footprint. However, because of the available energy supply resources (i.e., small batteries or inefficient energy harvesters), [184] WCS usually includes low-power MCUs (e.g., ARM Cortex-M or RISC-V PULP), which presents limited resources that can prevent executing data filtering tasks under real-time constraints [185]. To address this challenge, this Section proposes an optimized image compression algorithm tailored for a RISC-V multi-core processor, that allows to shrink the image size and the data to transmit. An optimized JPEG (Joint Photographic Experts Group) encoder is presented, which is based on Fast-DCT (FDCT) image compression algorithm, with an adaptive trade-off between energy consumption and image distortion. The solution is tailored for parallel fixed-point computing hardware and exploits DSP-oriented instructions included in the RISC-V extended ISA (Instruction Set Architecture) of PULP. When compared with a JPEG implementation on ARM Cortex-M4, this solution achieves a frame rate of 22 fps and it is 8× more energy-efficient, if running on the GAP8 processor, an eight cores embodiment of the PULP architecture. The algorithm is released as an open source project on GitHub[1].

Several hardware accelerators are available as standalone chips or add-on-IP blocks for system-on-chip integration [186]. However, the extra cost (in silicon area and/or bill of materials) for a hardware JPEG encoder may not be affordable in many application scenarios that require software JPEG compression. Since the 1990, FDCT algorithms for image compression have been intensively studied in the literature [187] to reduce the number of CPU instructions needed to operate on a standard block, an 8×8 matrix of pixels. Indeed, the image compression function based on the 2-D 8-point DCT is prevalent, which is typically the most computationally intensive. Among the various fast DCT algorithm proposed, the following four are the most common. The first fast DCT, proposed by Chen [183], has an excellent regular structure, but it requires as many as 16 multiplications for each 8-point block. Hou [188] proposed a recursive algorithm, with 12 multiplications and 29 additions. Although the number of operations is the same as other fast algorithms, it has the advantage of the smaller number of variables necessary for the execution. The function proposed by Loffler [189] involves 11 multiplications and 29 additions. Additionally, the authors proposed a parallel solution that simultaneously executes three multiplications. Finally, the algorithm proposed by Arai [190] features a simplification of the DCT processing. It requires only 5 multiplications and 29 additions.

---

[1] https://github.com/GreenWaves-Technologies/gap_sdk/tree/master/applications/jpeg_encoder

Moreover, it can be easily implemented with fixed-point operations, speeding up the code execution in the absence of a Floating Point Unit (FPU). The aforementioned works make clear that using an optimized DCT algorithm heavily decreases the number of operations required by the JPEG encoder and that a parallelized execution can be applied.

This work is based on Noritsuna, a JPEG encoder optimized for Cortex-M4 [191]. This implementation supports floating-point operation at low memory impact, but it is not tailored for real-time compression since it is based on a non-fast DCT algorithm (Table 7.1 – Cycles 1). To overcome this issue, the DCT algorithm is replaced with the Arai [190] FDCT implementation. However, the Noritsuna's algorithm implementation applies to individual 8 × 8 image blocks, hence demanding low L1 memory footprint and favoring a block-wise parallelization scheme for multi-core implementation. After an in-depth study, the application described in [192] was selected as a comparison for this work; indeed, it needs only 10 Mcycles (220 ms @ 48 MHz) to compress a QVGA grayscale frame, about 8 Kcycles/block, one of the best performance with a low-power ARM Cortex-M4. Similarly to the solution here described, this implementation exploits fast DCT, but it is optimized on Cortex-M4 architecture featuring an L1 scratchpad memory of 80 kB (with QVGA resolution), greater than the GAP8 cluster memory. Among other solutions, the authors in [187] describe an optimized firmware that needs 22-26 Mcycles to compress a 752 × 480 pixel in RGB format (≈9 Kcycles/block), whereas the paper in [182] requires 300 kcycles to process a single 8 × 8 block, with an average execution time of 9.207 ms on a Texas Instruments MSP430. The deployment in [182] uses up to 29 mJ to encode a single 128 × 128 picture.

7.2 JPEG algorithm: implementation and optimization

The original version of the firmware [191] is composed of the following steps: (*i*) generation of the header file; (*ii*) image decomposition into 8 × 8 pixel blocks, and if the overall dimensions are not multiple integers of 8, the missing blocks are padded with values calculated from the average value on the edges, then the level shifting is executed; (*iii*) application of the DCT to every block, followed by the quantization, and zigzag operations; (*iv*) Huffman; (*v*) writing back the compressed data into the L2 memory.

Since the GAP8 architecture is not equipped with an FPU, all the operations are implemented with a fixed-point representation. For this data type, it is needed to select in advance the number of bits dedicated to the integer and the fractional parts, and, depending on this choice, the JPEG encoder can achieve higher precision (increasing the number of fractional bits) or a broader dynamic range. An empirical evaluation individuates the best trade-off by selecting 15 bits for the fractional part and 16 bits for the integer part (16Q15). To quantitatively evaluate the differences between both representations, this work adopts as mean metrics the Peak Signal to Noise Ratio (PSNR) and the Mean Squared Error (MSE) since they are widely used in the scientific community as evaluation indexes in the field of image processing [193]. The 16Q15 representation covers the dynamical range required by the algorithm and increases the PSNR of 0.3%, and the MSE is practically unchanged concerning the floating-point original code. Table 7.1 – Cycles1 reports the GAP8 performance metric to run the JPEG implementation on a QVGA image (324 x 240). This initial version requires more than 130 M cycles, at 50 MHz, the frame conversion time is approximately 2.5 s. The latency breakdown individuates the DCT routine as the most demanding part from a computational point of view, as expected from the description of the firmware in [191]. Following the formula given in [191], we need 3,136 additions and 8,192 multiplications, meaning a considerable load on processors, especially RISC, where multiplications require greater use of resources. The optimization usually focuses on reducing the number of arithmetic operations to be performed during the DCT. Like most of the fast algorithms, the one proposed by Arai, Agui, and Nakajima [190] exploits the separability of the two-dimensional DCT and reduces it to the calculation of a one-dimensional DCT on eight elements for all the rows and subsequently for the columns. This algorithm is considered the fastest: it requires 29 additions and 5 multiplications for the DCT 1D and 464 additions and 144 multiplications for the 2D DCT on the 8 x 8 block. The JPEG encoder performance with AAN (Arai Arui Nakajama) DCT is presented in Table 7.1 - Cycles2. With this change, the major performance improvement was achieved, dropping the total number of cycles by 89%, mainly due to the relative reduction by 98% in the execution of the DCT. On the other hand, since the AAN algorithm approximates a standard DCT, it impacts the quality of the output image, increasing the MSE of about 86%. However, as shown in Figure 7.1, the image quality difference perceived from a human eye is negligible despite the MSE and PSNR indexes drop; hence the AAN DCT can be considered a suitable replacement in our JPEG encoder. The first (Cycles1) implementation (Noritsuna [191]), written following the theoretical definition of the 2D DCT, presents a complexity $O(n^4)$; the AAN instead reduced the complexity to $O(n \cdot log_2(n))$ motivating the notable latency reduction. A third optimization step for sequential execution is performed using the hardware features available on GAP8 SoC, such as the DSP-oriented extended-ISA instructions (built-in) and the single cycles access memory (L1). The built-in functions are extensions of the RISC-V instruction set, developed to speed up some computationally heavy operations. Among the most commonly used, the algorithm exploits the Multiply Accumulate (MAC) instructions, which multiply two variables and accumulate the partial sums, and the FIXED_MUL, which multiplies

Figure 7.1: Image quality comparison between FAST-DCT (AAN) and DCT algorithms.

two fixed-point variables in one single cycle. The final number of cycles required for an in-line execution is presented in Table 7.1 – Cycles3. To run the JPEG encoder on the GAP8 cluster, the algorithm steps are executed using the available 8 RISC-V cores. The initial section of the JPEG file header can be performed only once at the beginning of the program since it is fixed (Figure 7.2 – *(a) – Header Writing*). The rest of the workload is distributed among the cluster by letting any core operate on different image 8 × 8 blocks (Figure 7.2 – *(a) – Multi-core functions*). Indeed, during the compression of the pictures, it is sufficient writing to the output file (L2) the bytes containing only the information concerning the actual image starting from the byte following the last of the header (Figure 7.2 – *(a) – Footer writing*). The image blocks reading function can be easily performed in parallel, similarly to level shifting, discrete transform of cosines, zigzag reordering, and quantization tasks. Instead, the Huffman task operates on data produced by previous steps. Hence it is executed as a sequential task on a single core. In addition to this, the Huffman encoding does not have a predefined number of bits

Table 7.1: Number of cycles required to execute the JPEG algorithm on different implementations.

Functions	Cycles1	Cycles2	Cycles3	Parallel
DCT + Zig-Zag (8 × 8 block)	107,500	1,947	1,873	1,611
Quantization (8 × 8 block)	2,539	2,539	2,220	2,368
Huffman (8 × 8 block)	984	984	984	802
Total (all image)	130,147,237	13,072,581	6,092,954	2,307,672
MSE	53	100	100	100
PSNR [dB]	31	29	29	29
Speedup	-	10	21	56

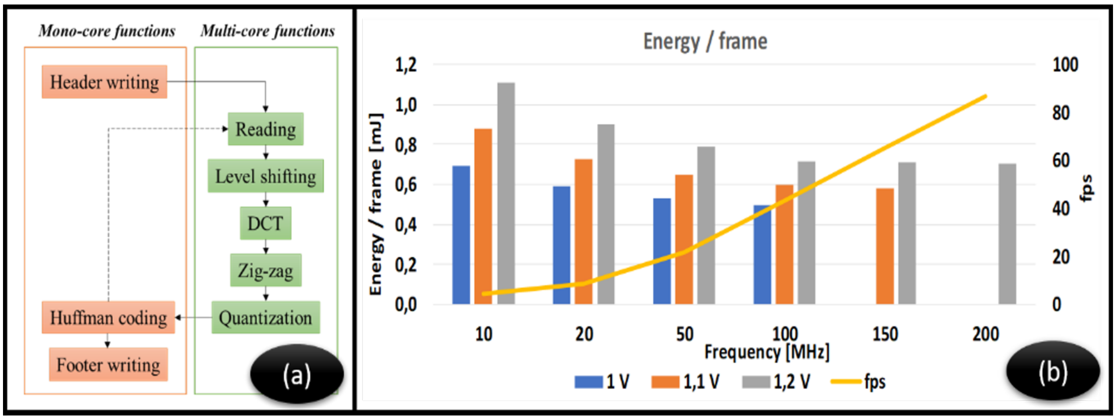


Figure 7.2: (a) JPEG sub-functions, the multi-core algorithms can be parallelized. Instead, the mono-core function must be executed sequentially; (b) Energy per frame and maximum fps compared to the cluster frequency and voltage.

needed to encode a symbol, but the output is of variable length. For this reason, it was considered necessary to separate this last step from parallel execution by executing it sequentially from a single cluster core. With the parallel execution of the firmware, the algorithm reaches 2,307,672 cycles (Table 7.1 – Parallel) and conversion time of about 45 ms @ 50 MHz, which corresponds to 22 frames per second. The speedup reached with a parallel execution is 2.64 with eight cores because the Huffman is executed sequentially.

7.2.1 Estimation of the system energy

In the case of multi-core execution, the highest energy efficiency is achieved at 1 V at the maximum frequency of 100 MHz (Figure 7.2 – (b)). At this operating point, the GAP8 compresses a frame with 0.495 mJ, while at 50 MHz the energy consumption results to be 0.532 mJ. With a supply voltage of 1.2 V, we can reach 200 MHz, compressing around 86 images per second, but the energy required for compression reaches 0.7 mJ per frame.

The performance metrics are analyzed with respect to a low-power MCU device, such as STM32L476G from STMicroelectronics, running the JPEG implementation of [191]. The reference MCU is an ultra-low-power platform based on a 32-bit ARM Cortex-M4 core capable of operating at a frequency up to 80 MHz. The STM32L476G, in RUN mode @ 48 MHz, consumes 18.29 mW. The obtained number of cycles is equal to 10,528,330 with a single QVGA image conversion. Hence the compression latency and the energy consumption are 0.22 s and 4.011 mJ per frame. In the same scenario, the proposed JPEG implementation, in conjunction with GAP8, reaches an execution time $\approx 5\times$ faster than an STM32L476, with an average energy consumption $8\times$ lower.

One of the most power-consuming tasks in WCS applications is to transfer the images acquired by the camera either to cloud servers or to personal gateways (e.g., a mobile phone) for low-latency feedback [182]. Consequently, wireless communication is an essential feature, although it is often the bottleneck both for the throughput and for the power budget of the entire system, considering that applications might need to stream images and videos continuously. In previous papers [25, 97], the joint challenge of communication energy minimization and maximization of the communication flexibility under several different connectivity scenarios was studied. The article [27] shows that to stream raw images, the Wi-Fi requires an average of 30 nJ/bit. In this chapter, the QVGA sensor generates an 80 kB/frame that a 20 fps produces up to 12.8 Mbps data, which needs 384 mJ to send 20 frames. On the other hand, using the described JPEG encoder, the compressed image uses only 3.8 kB, generating 608 kbps. The GAP8 needs 9.9 mJ, but the Wi-Fi energy decreases to 18 mJ with overall consumption of 27.9 mJ, which is an improvement of 14× in energy efficiency.

7.3 Ultra-low energy consumption pest detection for smart agriculture

In the last years, ICT techniques that can optimize the harvest of fruit and vegetables have gained momentum. In particular, over-population and the rise of the climate crisis necessitates smart technologies capable of minimizing the waste of resources and protecting the cultivations [194]. For example, biological invasions could have severe global consequences if not handled correctly, including ecological destruction and economic losses. Thus, if not tackled promptly, crop losses and pest control can be extremely expensive [195].

This Section presents a method to detect and identify dangerous insects in the field, to trigger an alarm to the farmer, and to minimize the damage caused by pests in the orchard. Currently, the technologies deployed in precision agriculture allow only to send raw images to the farmer to decide if and how to counteract [194]. The proposed system innovation combines sophisticated low energy sensors for the acquisition step, an efficient algorithm for the on-sensor analysis based on machine learning, with long-range communication. The system is designed for minimum power consumption; thus, it can also be energy autonomous thanks to a small-size energy harvesting circuit.

Adding intelligence to the nodes and shifting the decision of anomalies near the sensor permits faster decisions and actions that are the key to damage reduction. Moreover, the low cost of the platform, its non-intrusive size, and ultra-low power design permit the high scalability of this solution in vast orchards. The model developed in this project belongs to the newest generation of agricultural sensing and automation devices, monitoring and sending real-time messages.

The system consists of a trap that looks like a little hive as shown in Fig. 7.3, where a pheromone bait and a glue layer capture the attracted insects even at an early stage when their density in the filed is low. The farmer usually takes periodic inspections of the traps or mount a wireless camera that sends the captured pictures wirelessly for remote evaluation. This process is expensive and time consuming for the farmer. The proposed work detects the parasite presence thanks to a machine learning approach that triggers notifications and their position to the farmer, only when threats are recognized. The smart camera consists of an ultra-low-power smart camera with on-situ reconfigurable AI-capabilities. A multi-core architecture named PULP, implemented on a chip called GAP8 (see Section 2.2.2.1) realized for IoT-inspired applications, is used to process the images. One of the most power-consuming tasks in smart sensors for agriculture is transferring data and images acquired by the camera over long distances [28, 29]. Wireless

Figure 7.3: Prototype of the IoT neural network Codling Moth smart trap.

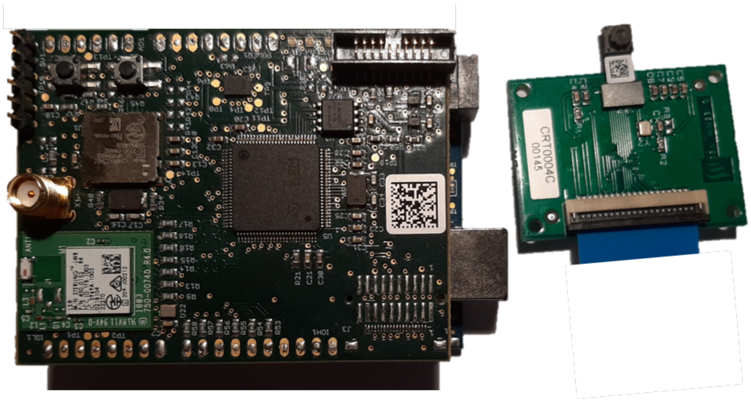


Figure 7.4: Smart camera device used in the application.

communication is an essential feature, but it is usually the bottleneck for data throughput and the power budget. This Section tackles the challenges by (*i*) minimizing the output data using a neural network-based detection algorithm, and (*ii*) we maximized the communication flexibility using an unlicensed wireless long-range protocol such as LoRaWAN. The smart camera board designed for the project is shown in Fig. 7.4.

7.4 Data analysis

The dataset generation started with a small set of row pictures, as shown in Fig. 7.5 (approximately 300). The dataset is divided into two classes: *codling moth* (Fig. 7.6 left) and *general insects* (Fig. 7.6 right). The smart camera makes use of a gray-scale QVGA CMOS sensor (Himax HM01B0) with a resolution of 244 x 324 pixels. Hence it needs 79 kB of RAM to be stored (L2 in GAP8). To decrease the neural network complexity and the average power consumption, each image is pre-processed by a stack

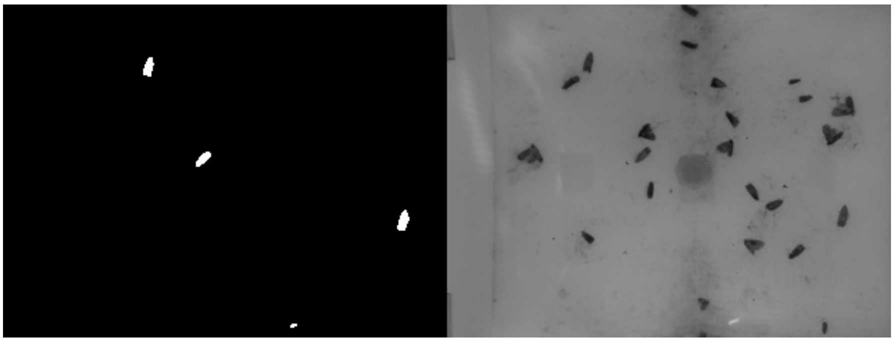


Figure 7.5: Left: processed image, white insects was not presented in previous frames. Right: raw picture.

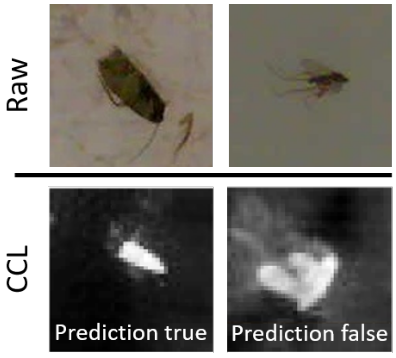


Figure 7.6: Cropped Codling Moth (left). Cropped general insect (right) and their respective data analysis results.

of morphological operators, determining if the acquired image contains new elements of interest.

First of all, an histogram equalization filter is applied to decrease the brightness difference between successive frames. It is usually implemented in C++ code [196] and needs floating-point operations. The optimized imported version running on GAP8 requires only 37 cycles-per-pixel (cpp) working on a fixed-point framework. On the original QVGA image the overall number of clocks is 2.9 M.

Afterward, a background subtraction algorithm is used to detect changes in image sequences and to highlight the information provided by the Himax. Background subtraction is often utilized for detecting moving objects from static cameras, but in our case, it serves to identify new trapped insects. In this work, the OpenCV MOG function is calibrated on the insect dataset, which is part of the Gaussian Mixture Model (GMM) foreground detection [197]. In-field results show that a history (α) of two images and the mixture depth (κ) of five are the best trade-off between complexity and accuracy in our specific application; therefore, these settings are used for the algorithm on GAP8.

The memory impact of the GMM is not negligible. It needs enough space to store the foreground history, which is two QVGA images and its associated statistical parameters.

In fact, each pixel must include all the representative parameters of a Gaussian, i.e., weight, mean, and variance, as floating-point variables. The size of the aforementioned structure, nominated as *Gauss*, is calculated as Eq. 7.1.

$$GMM_{float} = Gauss_{size} \cdot 244 \cdot 324 \cdot \alpha \cdot \kappa = 9.5\ MB. \tag{7.1}$$

A lightweight version of the standard OpenCV MOG function for GAP8 is implemented by developing the firmware using only fixed-point math. The code execution is performed using a 32-bit depth, with Q16.16 number format, while the *Gauss* structure is considered as an 8-bit variable with different fractional settings. Weights and mean are represented respectively as Q1.7 and Q8.0 while the variance is in Q6.2 format. With this method, the GMM requires only $GMM_{fixed} = 1.2\ MB$, 5× less than GMM_{float} and 392 cpp executed on the GAP8 FC. Compared with the standard OpenCV MOG function [197], the output degradation is below 1%; the error is mainly composed of isolated pixels not recognized as part of the background. The execution of GMM is the most power-hungry, requiring 60% of the overall energy. This algorithm is sequentially running on the FC and will be further parallelized in future works.

The next step consists of a cascade of morphological operators to remove the noise after the GMM and cut the non-interesting objects, those too big or too small to be a general insect. Two iterations of Closing (morph close [197]) and one Opening (morph open [197]) operators make the processed image ready for the last processing task. They need an average of 27.5 cpp.

The last task consists of a connected component (CCL) function crop and saves (in L3) only the sub-parts (if any) of the original images that are interesting for the application, producing images of 40x40 pixels. This last layer of the stack requires 2.8 cpp.

Table 7.2 summarizes the execution time and workload for the whole stack, which requires 40.5 M Cycles and 2.8 mJ. Moreover, Fig. 7.7 displays the image acquisition pipeline described above.

Table 7.2: Data Analysis Stack

Step	M Cycles	Time [ms]	Energy [mJ]
Hist. Eq.	2.9	58	0.20
GMM	30.9	618	2.10
Closing	4.3	86	0.29
Opening	2.2	44	0.15
CCL	0.2	4	0.01
Total	40.5	810	2.8

Time calculated considering the FC running at 50 MHz

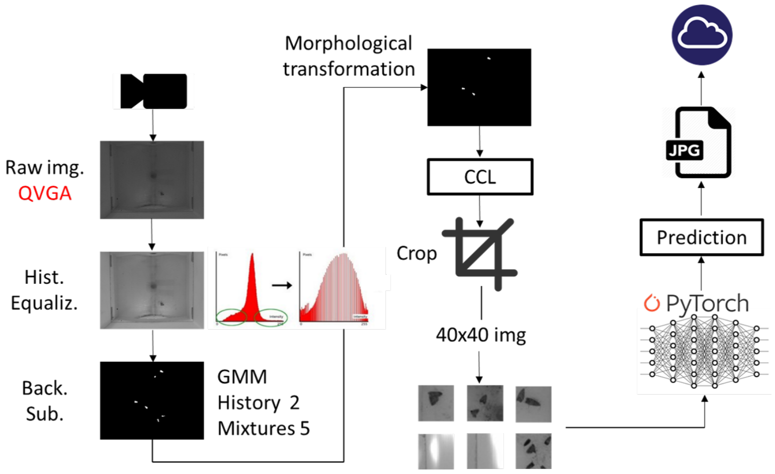


Figure 7.7: Image acquisition pipeline.

7.4.1 NEMO

In a Quantized Neural Network (QNN), all tensors $\mathbf{t}$ (e.g., inputs $\mathbf{x}$, outputs $\mathbf{y}$, and weights $\mathbf{w}$) are real-valued, but they can assume only a countable set of values (fake discretization) and are determined only in a specific range $[\alpha_{\mathbf{t}}; \beta_{\mathbf{t}})$. It is possible to map these discretized variables into purely integer ones, which are images of the discretized tensor. The following section assumes to start from a QNN already discretized using linear, uniform per-layer quantization: the N-bit integer image $\hat{\mathbf{t}}$ of tensor $\mathbf{t}$ is linked to its quantized counterpart through a bijective mapping, as shown in Eq. 7.2, where $\epsilon_{\mathbf{t}} = (\beta_{\mathbf{t}} - \alpha_{\mathbf{t}})/(2^N - 1)$.

$$\mathbf{t} = \alpha_{\mathbf{t}} + \epsilon_{\mathbf{t}} \cdot \hat{\mathbf{t}}. \tag{7.2}$$

The $\epsilon_{\mathbf{t}}$ value describes the *quantum*, the smallest number represented in the quantized tensor.

Each QNN is normally composed of a sequence of three operators: Linear/Convolutional, Batch-Normalization and Quantization/ Activation. Without loss of generality[2], it is possible to consider for all the inputs that $\alpha_{\mathbf{x}} = \alpha_{\mathbf{y}} = 0$. Using Eq 7.2, all variables are mapped in the integer domain. Hence it is mathematically valid to re-map linear (LN) and Batch-Normalization (BN) operators.

$$LIN: \ \varphi = \sum_n \mathbf{W}_{m,n} \cdot \mathbf{X}_n \Leftrightarrow \hat{\varphi} = \sum_n \widehat{\mathbf{W}_{m,n}} \cdot \widehat{\mathbf{X}_n}, \tag{7.3}$$

$$BN: \ \varphi' = k \cdot \varphi + \lambda \Leftrightarrow \hat{\varphi}' = \hat{k} \cdot \hat{\varphi} + \hat{\lambda}. \tag{7.4}$$

The dot operation in Eq 7.3 shrinks the quantum used to present $\hat{\varphi}$, which is described by $\epsilon_{\varphi} = \epsilon_{\mathbf{w}} \cdot \epsilon_{\mathbf{x}}$. Hence, the integer output represents the LN ($\hat{\varphi}$) with augmented

---

[2] If the original activation is a ReLU, then the QNN automatically satisfies this condition; otherwise, it can be transformed to satisfy it.

precision (standard of 32 b on GAP8) with respect to its input. Then, $\hat{\varphi}$ is re-quantized when the accumulation ends. A similar conversion is applied to BN and its output ($\hat{\varphi}'$).

The final operator provides a non-linear activation (essential in any neural network) and collapses the accumulator into a lower precision variable.

$$ACT: \ \hat{\mathbf{y}} = m \cdot \hat{\varphi}' \gg d; \ m = \left[\frac{\epsilon_{\varphi'} \cdot 2^d}{\epsilon_{\mathbf{y}}}\right]. \tag{7.5}$$

In Eq 7.5, d is an integer that guarantees a sufficient representation of $\epsilon_{\varphi}/\epsilon_{\mathbf{y}}$, it is experimentally chosen during the quantization process. Finally, by the use of 7.2, it is possible to execute a QNN on a constrained device, such as GAP8, using only integer operators. NEMO (NEural Minimization for pytOrch) usually targets networks using 8-bit quantization for both $\hat{\mathbf{w}}$ and $\hat{\mathbf{x}}$ $\hat{\mathbf{y}}$, while $\hat{\varphi}$, $\hat{\varphi}'$, $\hat{k}$, $\hat{\lambda}$, m, and d variables are usually 32-bit bit-width.

7.4.2 DORY

The deployment of DNNs on IoT end-nodes at the extreme edge is still a critical technology challenge to support pervasive deep learning applications. Low power and low-cost MCUs have limited on-chip memory to reduce area overheads and increase energy efficiency, requiring explicit memory transfers between internal and external data cheches. Running complex DNNs on these constrained devices requires smart topology-dependent tiling and double-buffering because the DNN does not fit into the internal RAM. DORY (Deployment Oriented to memoRY) is an automatic tool to deploy DNNs on low-cost MCUs with typically less than 1 MB of on-chip RAM. It maximizes internal cache memory utilization under the constraints imposed by each DNN layer. Then, it automatically generates ANSI C code, managing DMA transfers and algorithms. Furthermore, DORY augments the computing formulation with heuristics promoting performance-effective tile sizes, aiming to maximize speed and decrease the memory footprint. DORY was designed and tested GreenWaves Technologies GAP8[3], described in Section 2.2.2.1. On GAP8 [198], DORY achieves up to 2.5× better MAC/cycle than the GreenWaves proprietary software solution and 18.1× better than the state-of-the-art result on an STM32F746. By using DORY, GAP8 can perform end-to-end inference of a 1.0-MobileNet network consuming 63 pJ/MAC on average, 15.4× better than an STM32F746.

---

[3] GAP8: https://greenwaves-technologies.com/

7.5 Neural network

To correctly analyze an image of 40 × 40 pixels, a machine learning approach is used. A Convolutional Neural network has been specifically trained for this application, named Smart Agriculture Neural Network (SANN), see Fig 7.7.

The SANN topology was inspired by CIFAR10 networks [199] and was reduced in size and complexity to minimize the bare image processing time using NEMO[4] and DORY[5] [198]. The SANN is composed of four successive convolutional layers and four fully-connected layers, in combination with batch normalization and ReLu activation functions.

It is trained on the small available dataset featuring a 97% training accuracy and 93% validation accuracy. These results are also confirmed after the quantization procedure (NEMO), where the weights and coefficients are represented in a fixed-point 8-bit format. Tests on GAP8 assess the computational complexity to 2.9 MCycles, considering the execution on the 8-cores cluster. The equivalent energy reaches 0.66 mJ; hence the overall energy consumption amounted to 3.5 mJ. Finally, if a parasite is detected, GAP8 encodes the original image in a JPEG format [61] to decrease the data size, and successively, send it remotely through LoRaWAN. By comparison, the average energy per bit of LoRaWAN is 2.2 mJ [200], which is equivalent to the whole data analysis in Section 7.4. In this application, the edge-computing paradigm can save energy up to five orders of magnitude.

7.6 Summary

In previous sections, an optimized JPEG encoder based on the FDCT and a *mJ*-class smart camera with an embedded tiny machine learning are presented, which are parallel executed on GAP8, a multi-core RISC-V SoC. These two case studies are the perfect example of edge computing, in which the data is compressed, extracting the most relevant features, directly on the sensor node. Moreover, energy projections and in-field test demonstrate the effective performance improvements of a low-power platform, increasing the battery lifetime and decreasing the network traffic.

In Section 7.1, the encoder can reach up to 86 fps @ 200 MHz, but at 100 MHz the MCU requires only 0.495 mJ to compress a frame, reaching the best trade-off between the compression rate (46 fps) and the energy consumption. When compared with a JPEG

---

[4] NEMO: https://github.com/pulp-platform/nemo

[5] DORY: https://github.com/pulp-platform/dory

implementation on ARM Cortex-M4 (48 MHz), this solution (@ 50 MHz) achieves a frame rate 4.8× higher and requires 8 times less energy to encode a single image. Instead, if compared to Noritsuna [191], this solution features 56× lower number of clock cycles. Lastly, the JPEG encoder was exploited in a real deployment, a QVGA sensor with a Wi-Fi module. In this application, the encoder can reduce the system energy up to 14× at 20 fps with respect to stream raw images through a WiFi connection.

Lastly, Section 7.3 presents a mJ-class smart camera with an embedded tiny machine learning model trained for precision agriculture services. The camera identifies the pests of apple in the orchards and triggers an alarm to the farmer. The extremely low power consumption permits the smart-trap to operate without any maintenance for years, using a LPWAN to promptly transmit reports to several kilometers. An optimized OpenCV based framework was specifically developed to archieve this challenging result.

Chapter 8

Conclusions

8.1 Thesis contributions and achievements

Aiming at an even more connected and burgeoning world, this thesis spans the IoT framework analyzing the most recent trends and technologies, providing an overview of main open challenges and limitations. Structured as a development and innovation guideline, the sequence of discussed arguments illustrates the five principal blocks forming a standard sensor node: transducers, power management, edge computing, and low power wide area network protocols. Nowadays, IoT has reached an unprecedented spread level and utilization in almost any industrial and healthcare application. Moreover, it is opening new scenarios to improve general wellness; indeed, automation and continuous monitoring processes decrease the working time and hazardous situations, such as the massive contribution provided from SHM installations, which can drastically reduce the number of collapsed buildings and workplace accidents. However, the increasing diffusion of IoT is generating unexampled technical and social challenges that must be solved every day, since they are continuously changing following the user and market requests.

The conventional trend is the deployment of battery supplied sensor nodes and actuators, directly or indirectly connected to the central node, a cloud, or a server [1, 3]. This generates an implicit requirement about the battery lifetime and the internal processing and remote communication energy. Researchers and developers bypass the limited energy resource by applying a power reduction mechanism through careful hardware planning. However, reducing the average apparatus current is often insufficient to comply with the most stringent requirements demanding $>$ 10 years of fully functional operativity. Hence other methods are taking place to enhance the device longevity, such as applying energy harvesting techniques, decreasing the wireless transmission energy by reducing the energy per bit. Moreover, it is possible to trim the number of remote transmissions through the edge-computing and, on the other side, designing novel low-power low-cost sensors. While conventional battery power is the most apparent solution for IoT applications, it is not always economical. Also, there is concern about the cost of new long-life batteries and the person-hours required to maintain them. Even rechargeable batteries, which need to be replaced periodically, have a non-negligible negative impact on the environment. As the IoT market becomes more environmentally conscious, this has become a serious concern. Finally, access to continuous power from the power grid is not always achievable. Whether it is the nature of the deployment, or the cost of connecting devices to a grid, accessing constant power is a recurring challenge for many IoT systems. In addition to employing energy-harvesting techniques that capture energy from the surrounding environment, IoT device designers can use protocols that save power. Three standards of low-power-consumption procedures are power-saving mode (PSM), Extended Discontinuous Reception (eDRX), and wake-up signals. Most cellular and LPWAN technologies turn off their module when the cellular network connection is not required, but reconnecting to the network requires power and time. The Power Saving Mode (PSM), on the other hand, exploits network connection timers to reduce IoT power consumption. Finally, wake-up radio allows IoT platforms to remain asleep and not periodically check the ether for incoming signals, which is especially useful for devices that do not need to communicate for long periods.

In Chapter 2, an overview of the IoT framework and the structure of a typical sensor node is proposed. It aims to provide basic knowledge to read and understand all the topics and design choices presented during the following discussion. Considerations and technology scouting about the state of the art technology show that *i*) achieving more than 10 years of operation for a device is still a challenge from a hardware point of view, *ii*) sensors and transducer are fundamental to provide accurate measurements at the lowest power consumption, and that they often heavily impact the final price, *iii*) wireless protocols must be optimized to reduce the communication redundancy, the latency and to support many thousands of devices in crowded communication channels, also keeping the energy per bit as low as possible, *iv*) edge computing is essential to decrease the communication latency and consumption, opening even new application scenarios.

In Chapter 3, six wireless protocols are investigated, analyzed, and measured in controlled environments and real deployments. For this thesis, a selection of the most promising and commercially used standards converges to LoRa, NB-IoT, UWB, WUR, WiFi, Bluetooth. Since selecting the best communication protocol for every application is still an open issue, Chapter 3 aims to provide a detailed overview of the real throughput and power consumption, which are often essential to justify a specific design choice. Moreover, a dedicated optimisation work concern *i*) LoRaWAN lightweight time synchronisation for heterogeneous networks, *ii*) Slotted-Aloha MAC working on top of the LoRaWAN framework, a plug-and-play solutions suitable for commercial products, *iii*) NB-IoT power classification and estimation under variable operations conditions, *iv*) a detailed comparison between LoRaWAN and NB-IoT for industrial and SHM applications, *v*) Extended LoRaWAN, a custom communication protocol designed to double the coverage of a standard LoRa node, *vi*) UWB description, specifically focused on Decawave DW1000.

Chapter 4 presents a specific assessment about the UWB together with IoT and autonomous drones. This niche but challenging topic aims to overcome connectivity limitation in scarcely populated areas and harsh environments. Main subjects are *i*) optimization of the UWB ranging protocol for high-precision distance estimation between moving objects and UAVs, supporting in parallel high-speed data transfer, *ii*) decreasing and assessing the UWB power consumption by the use of duty cycle and WUR mechanisms.

In Chapter 5, the design of a new low cost and low power transducer is addressed. To overcome the stringent limitations in one of the harshest applications, SHM, it was needed to design a specific transducer capable of 1 μm precision at $<$ 5€ cost. The final device, named *Crackmeter* was *i*) tested, *ii*) characterized varying temperature and simulating aging, *iii*) assembled with a custom plastic enclosure made of polycarbonate, *iv*) put together with a low power electronics to make a fully functional IoT sensor node. Finally, the *Crackmeter* is today a commercial product sold by Sacertis s.r.l.[1]

In Chapter 6, two design examples are presented, showing how to increase the battery lifetime by applying hardware-specific optimizations. Considering the use of a sensor fusion approach, in which multiple sensors are working at a different frequency and for numerous purposes, the following observations are valid *i*) split the power supply line into multiples clusters, controlled by the MCU, *ii*) avoid, when and where it is possible, the use of passive pull-up/pull-down, *iii*) ultra-low power and asynchronous wake-ups are always preferable than polling or duty-cycled approaches, *iv*) firmware control is fundamental to keep the MCU is the lowest power consumption state.

In Chapter 7, the paradigm of edge computing is exploited in two different scenarios, an image encoder (JPEG) and a fully working framework to recognize dangerous insects in agricultural environments. Since the deployment of complex algorithms at the extreme edge of the IoT structure is still an unsolved challenge, the design was based on a PULP platform, the state of the art technology for the high computational task at a lower energy cost. This chapter shows that *i*) a complex algorithm such as morphological

---

[1] http://www.sacertis.com/

operators and neural networks can be deployed in a constrained and low power device, *ii*) the edge computing can decrease the energy consumption up to five orders of magnitude compared to streaming all the data directly to the cloud, *iii*) despite limited resources and approximating algebra, the accuracy of the embedded-algorithms is comparable with server responses.

Bibliography

[1] Wei Yu, Fan Liang, Xiaofei He, William Grant Hatcher, Chao Lu, Jie Lin, and Xinyu Yang. A survey on the edge computing for the internet of things. *IEEE access*, 6:6900–6919, 2017.

[2] Farzad Samie, Lars Bauer, and Jörg Henkel. Iot technologies for embedded computing: A survey. In *2016 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ ISSS)*, pages 1–10. IEEE, 2016.

[3] Somayya Madakam, Vihar Lake, Vihar Lake, Vihar Lake, et al. Internet of things (iot): A literature review. *Journal of Computer and Communications*, 3(05):164, 2015.

[4] Michele De Donno, Koen Tange, and Nicola Dragoni. Foundations and evolution of modern computing paradigms: Cloud, iot, edge, and fog. *Ieee Access*, 7:150936–150948, 2019.

[5] Partha Pratim Ray. A survey on internet of things architectures. *Journal of King Saud University-Computer and Information Sciences*, 30(3):291–319, 2018.

[6] Halil Yetgin, Kent Tsz Kan Cheung, Mohammed El-Hajjar, and Lajos Hanzo Hanzo. A survey of network lifetime maximization techniques in wireless sensor networks. *IEEE Communications Surveys & Tutorials*, 19(2):828–854, 2017.

[7] Prabal Verma and Sandeep K Sood. Fog assisted-iot enabled patient health monitoring in smart homes. *IEEE Internet of Things Journal*, 5(3):1789–1796, 2018.

[8] Arsalan Mosenia, Susmita Sur-Kolay, Anand Raghunathan, and Niraj K Jha. Wearable medical sensor-based system design: A survey. *IEEE Transactions on Multi-Scale Computing Systems*, 3(2):124–138, 2017.

[9] Muhammad Shoaib Farooq, Shamyla Riaz, Adnan Abid, Kamran Abid, and Muhammad Azhar Naeem. A survey on the role of iot in agriculture for the implementation of smart farming. *IEEE Access*, 7:156237–156271, 2019.

[10] Yongyi Ran, Xin Zhou, Pengfeng Lin, Yonggang Wen, and Ruilong Deng. A survey of predictive maintenance: Systems, purposes and approaches. *arXiv preprint arXiv:1912.07383*, 2019.

[11] Prerit Datta, Akbar Siami Namin, and Moitrayee Chatterjee. A survey of privacy concerns in wearable devices. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 4549–4553. IEEE, 2018.

[12] Fotios Zantalis, Grigorios Koulouras, Sotiris Karabetsos, and Dionisis Kandris. A review of machine learning and iot in smart transportation. *Future Internet*, 11 (4):94, 2019.

[13] Bigi Varghese Philip, Tansu Alpcan, Jiong Jin, and Marimuthu Palaniswami. Distributed real-time iot for autonomous vehicles. *IEEE Transactions on Industrial Informatics*, 15(2):1131–1140, 2018.

[14] Michele Magno, Faycal Ait Aoudia, Matthieu Gautier, Olivier Berder, and Luca Benini. Wulora: An energy efficient iot end-node for energy harvesting and heterogeneous communication. In *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2017*, pages 1528–1533. IEEE, 2017.

[15] Jiao Zhang, Xiping Hu, Zhaolong Ning, Edith C-H Ngai, Li Zhou, Jibo Wei, Jun Cheng, and Bin Hu. Energy-latency tradeoff for energy-aware offloading in mobile edge computing networks. *IEEE Internet of Things Journal*, 5(4):2633–2645, 2017.

[16] Shivangi Vashi, Jyotsnamayee Ram, Janit Modi, Saurav Verma, and Chetana Prakash. Internet of things (iot): A vision, architectural elements, and security issues. In *2017 international conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, pages 492–496. IEEE, 2017.

[17] The national intelligence council sponsor workshop. Disruptive civil technologies. six technologies with potential impacts on us interests out to 2025. URL `https://fas.org/irp/nic/disruptive.pdf`.

[18] Abdullah Kurtoglu, Joan Carletta, and Kye-Shin Lee. Energy consumption in long-range linear wireless sensor networks using lorawan and zigbee. In *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 1163–1167. IEEE, 2017.

[19] Tobias Zillner and Sebastian Strobl. Zigbee exploited: The good, the bad and the ugly. *Black Hat–2015 [ ].– : https://www. blackhat. com/docs/us-15/materials/us-15-Zillner-ZigBee-Exploited-The-Good-The-Bad-And-The-Ugly. pdf (: 21.03. 2018)*, 2015.

[20] Nicolas Sornin, Miguel Luis, Thomas Eirich, Thorsten Kramp, and Olivier Hersent. Lorawan specification. *LoRa alliance*, 2015.

[21] Xavier Vilajosana, Thomas Watteyne, Tengfei Chang, Mališa Vučinić, Simon Duquennoy, and Pascal Thubert. Ietf 6tisch: A tutorial. *IEEE Communications Surveys & Tutorials*, 22(1):595–615, 2019.

[22] Wazir Zada Khan, Ejaz Ahmed, Saqib Hakak, Ibrar Yaqoob, and Arif Ahmed. Edge computing: A survey. *Future Generation Computer Systems*, 97:219–235, 2019.

[23] Alessio Burrello, Alex Marchioni, Davide Brunelli, and Luca Benini. Embedding principal component analysis for data reduction in structural health monitoring on low-cost iot gateways. In *Proceedings of the 16th ACM International Conference on Computing Frontiers*, pages 235–239, 2019.

[24] T. Polonelli, D. Brunelli, and L. Benini. Slotted aloha overlay on lorawan - a distributed synchronization approach. In *2018 IEEE 16th International Conference on Embedded and Ubiquitous Computing (EUC)*, pages 129–132, 2018.

[25] Tommaso Polonelli, Davide Brunelli, Achille Marzocchi, and Luca Benini. Slotted aloha on lorawan-design, analysis, and deployment. *Sensors*, 19(4):838, 2019.

[26] Tommaso Polonelli, Davide Brunelli, Marco Guermandi, and Luca Benini. An accurate low-cost crackmeter with lorawan communication and energy harvesting capability. In *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)*, volume 1, pages 671–676. IEEE, 2018.

[27] Tommaso Polonelli, Davide Brunelli, Alberto Girolami, Gerardo Nahuel Demmi, and Luca Benini. A multi-protocol system for configurable data streaming on iot healthcare devices. In *2019 IEEE 8th international workshop on advances in sensors and interfaces (IWASI)*, pages 112–117. IEEE, 2019.

[28] Massimo Ballerini, Tommaso Polonelli, Davide Brunelli, Michele Magno, and Luca Benini. Nb-iot vs. lorawan: An experimental evaluation for industrial applications. *IEEE Transactions on Industrial Informatics*, 2020.

[29] Massimo Ballerini, Tommaso Polonelli, Davide Brunelli, Michele Magno, and Luca Benini. Experimental evaluation on nb-iot and lorawan for industrial and iot applications. In *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, volume 1, pages 1729–1732. IEEE, 2019.

[30] Tommaso Polonelli, Davide Brunelli, Andrea Bartolini, and Luca Benini. A lorawan wireless sensor network for data center temperature monitoring. In *International Conference on Applications in Electronics Pervading Industry, Environment and Society*, pages 169–177. Springer, 2018.

[31] Tommaso Polonelli, Daniele Battistini, Manuele Rusci, Davide Brunelli, and Luca Benini. An energy optimized jpeg encoder for parallel ultra-low-power processing-platforms. In *International Conference on Applications in Electronics Pervading Industry, Environment and Society*, pages 125–133. Springer, 2019.

[32] Tommaso Polonelli, Yuan Qin, Eric M Yeatman, Luca Benini, and David Boyle. A flexible, low-power platform for uav-based data collection from remote sensors. *IEEE Access*, 8:164775–164785, 2020.

[33] Davide Brunelli, Tommaso Polonelli, and Luca Benini. Ultra-low energy pest detection for smart agriculture. In *2020 IEEE Sensors*, pages 1–4. IEEE, 2020.

[34] Mustafa Kocakulak and Ismail Butun. An overview of wireless sensor networks towards internet of things. In *2017 IEEE 7th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 1–6. IEEE, 2017.

[35] Anuradha Pughat and Vidushi Sharma. A review on stochastic approach for dynamic power management in wireless sensor networks. *Human-Centric Computing and Information Sciences*, 5(1):4, 2015.

[36] Shancang Li, Li Da Xu, and Xinheng Wang. Compressed sensing signal and data acquisition in wireless sensor networks and internet of things. *IEEE Transactions on Industrial Informatics*, 9(4):2177–2186, 2012.

[37] Kofi Sarpong Adu-Manu, Nadir Adam, Cristiano Tapparello, Hoda Ayatollahi, and Wendi Heinzelman. Energy-harvesting wireless sensor networks (eh-wsns) a review. *ACM Transactions on Sensor Networks (TOSN)*, 14(2):1–50, 2018.

[38] Ema Teixeira, Paulo Pedreiras, and Alexandre Mota. Low power wsn protocol for smart green homes. In *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1411–1414. IEEE, 2019.

[39] Chongfeng Wei and Xingjian Jing. A comprehensive review on vibration energy harvesting: Modelling and realization. *Renewable and Sustainable Energy Reviews*, 74:1–18, 2017.

[40] Antonino Proto, Marek Penhaker, Silvia Conforto, and Maurizio Schmid. Nanogenerators for human body energy harvesting. *Trends in biotechnology*, 35(7): 610–624, 2017.

[41] Yang Zhang, Hong Gao, Siyao Cheng, and Jianzhong Li. An efficient eh-wsn energy management mechanism. *Tsinghua Science and Technology*, 23(4):406–418, 2018.

[42] Neha Garg and Ritu Garg. Energy harvesting in iot devices: A survey. In *2017 International Conference on Intelligent Sustainable Systems (ICISS)*, pages 127–131. IEEE, 2017.

[43] Shancang Li, Li Da Xu, and Shanshan Zhao. The internet of things: a survey. *Information Systems Frontiers*, 17(2):243–259, 2015.

[44] Sukun Kim, Shamim Pakzad, David Culler, James Demmel, Gregory Fenves, Steve Glaser, and Martin Turon. Wireless sensor networks for structural health monitoring. In *Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 427–428, 2006.

[45] Ali Jameel Al-Mousawi and Haider K AL-Hassani. A survey in wireless sensor network for explosives detection. *Computers & Electrical Engineering*, 72:682–701, 2018.

[46] Milica Pejanović Đurišić, Zhilbert Tafa, Goran Dimić, and Veljko Milutinović. A survey of military applications of wireless sensor networks. In *2012 Mediterranean conference on embedded computing (MECO)*, pages 196–199. IEEE, 2012.

[47] Pasquale Daponte, Francesco Lamonaca, Francesco Picariello, Luca De Vito, Gianluca Mazzilli, and Ioan Tudosa. A survey of measurement applications based on iot. In *2018 Workshop on Metrology for Industry 4.0 and IoT*, pages 1–6. IEEE, 2018.

[48] Sampo Tuukkanen and Satu Rajala. A survey of printable piezoelectric sensors. In *2015 IEEE SENSORS*, pages 1–4. IEEE, 2015.

[49] Krittika Goyal and Akhil Gupta. A literature survey on different types of pulse based sensor for acquisition of pulse. *International Journal of Control Theory and Applications*, 9(10):361–365, 2016.

[50] Valentina Zega, Alessandro Nastro, Marco Ferrari, Raffaele Ardito, Vittorio Ferrari, and Alberto Corigliano. Design, fabrication and experimental validation of a mems periodic auxetic structure. *Smart Materials and Structures*, 28(9):095011, 2019.

[51] Manuele Rusci, Davide Rossi, Michela Lecca, Massimo Gottardi, Elisabetta Farella, and Luca Benini. An event-driven ultra-low-power smart visual sensor. *IEEE Sensors Journal*, 16(13):5344–5353, 2016.

[52] Daniele Palossi, Francesco Conti, and Luca Benini. An open source and open hardware deep learning-powered visual navigation engine for autonomous nano-uavs. In *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, pages 604–611. IEEE, 2019.

[53] Daniele Palossi, Antonio Loquercio, Francesco Conti, Eric Flamand, Davide Scaramuzza, and Luca Benini. A 64-mw dnn-based visual navigation engine for autonomous nano-drones. *IEEE Internet of Things Journal*, 6(5):8357–8371, 2019.

[54] Inc. Himax Imaging. Hm01bo 320x240 qvgacmos image sensor. URL `https://www.himax.com.tw/products/cmos-image-sensor/image-sensors/hm01b0/`.

[55] Michele Magno, Tommaso Polonelli, Luca Benini, and Emanuel Popovici. A low cost, highly scalable wireless sensor network solution to achieve smart led light control for green buildings. *IEEE Sensors Journal*, 15(5):2963–2973, 2014.

[56] Michele Magno, Angelo D'Aloia, Tommaso Polonelli, Lorenzo Spadaro, and Luca Benini. Shelmet: an intelligent self-sustaining multi sensors smart helmet for bikers. In *International Conference on Sensor Systems and Software*, pages 55–67. Springer, 2016.

[57] Tommaso Polonelli, Davide Brunelli, Andrea Bartolini, and Luca Benini. A lorawan wireless sensor network for data center temperature monitoring. In *International Conference on Applications in Electronics Pervading Industry, Environment and Society*, pages 169–177. Springer, 2018.

[58] Eric Flamand, Davide Rossi, Francesco Conti, Igor Loi, Antonio Pullini, Florent Rotenberg, and Luca Benini. Gap-8: A risc-v soc for ai at the edge of the iot. In *2018 IEEE 29th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, pages 1–4. IEEE, 2018.

[59] Eric Hamilton. What is edge computing: The network edge explained. *Cloudwards. Retrieved*, 3:18–20, 2019.

[60] Massimo Merenda, Carlo Porcaro, and Demetrio Iero. Edge machine learning for ai-enabled iot devices: A review. *Sensors*, 20(9):2533, 2020.

[61] Tommaso Polonelli, Daniele Battistini, Manuele Rusci, Davide Brunelli, and Luca Benini. An energy optimized jpeg encoder for parallel ultra-low-power. *Applications in Electronics Pervading Industry, Environment and Society: APPLEPIES 2019*, 627:125, 2020.

[62] Khaled B Letaief, Wei Chen, Yuanming Shi, Jun Zhang, and Ying-Jun Angela Zhang. The roadmap to 6g: Ai empowered wireless networks. *IEEE Communications Magazine*, 57(8):84–90, 2019.

[63] Qiang Xu, Todd Mytkowicz, and Nam Sung Kim. Approximate computing: A survey. *IEEE Design & Test*, 33(1):8–22, 2015.

[64] Francesco Conti, Robert Schilling, Pasquale Davide Schiavone, Antonio Pullini, Davide Rossi, Frank Kağan Gürkaynak, Michael Muehlberghuber, Michael Gautschi, Igor Loi, Germain Haugou, et al. An iot endpoint system-on-chip for secure and energy-efficient near-sensor analytics. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 64(9):2481–2494, 2017.

[65] Francesco Conti, Manuele Rusci, and Luca Benini. The memory challenge in ultra-low power deep learning. In *NANO-CHIPS 2030*, pages 323–349. Springer, 2020.

[66] Hongxing Gao, Wei Tao, Dongchao Wen, Tse-Wei Chen, Kinya Osa, and Masami Kato. Ifq-net: Integrated fixed-point quantization networks for embedded vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 607–615, 2018.

[67] Yu-Hsin Chen, Joel Emer, and Vivienne Sze. Eyeriss: A spatial architecture for energy-efficient dataflow for convolutional neural networks. *ACM SIGARCH Computer Architecture News*, 44(3):367–379, 2016.

[68] Angelo Garofalo, Giuseppe Tagliavini, Francesco Conti, Davide Rossi, and Luca Benini. Xpulpnn: accelerating quantized neural networks on risc-v processors through isa extensions. In *2020 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 186–191. IEEE, 2020.

[69] Alessio Burrello, Angelo Garofalo, Nazareno Bruschi, Giuseppe Tagliavini, Davide Rossi, and Francesco Conti. Dory: Automatic end-to-end deployment of real-world dnns on low-cost iot mcus. *arXiv preprint arXiv:2008.07127*, 2020.

[70] Parag S Chandakkar, Yikang Li, Pak Lun Kevin Ding, and Baoxin Li. Strategies for re-training a pruned neural network in an edge computing paradigm. In *2017 IEEE International Conference on Edge Computing (EDGE)*, pages 244–247. IEEE, 2017.

[71] Rih-Teng Wu, Ankush Singla, Mohammad R Jahanshahi, Elisa Bertino, Bong Jun Ko, and Dinesh Verma. Pruning deep convolutional neural networks for efficient edge computing in condition assessment of infrastructures. *Computer-Aided Civil and Infrastructure Engineering*, 34(9):774–789, 2019.

[72] Usman Raza, Parag Kulkarni, and Mahesh Sooriyabandara. Low power wide area networks: An overview. *IEEE Communications Surveys & Tutorials*, 19(2):855–873, 2017.

[73] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. Internet of things (iot): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7):1645–1660, 2013.

[74] Juha Petajajarvi, Konstantin Mikhaylov, Antti Roivainen, Tuomo Hanninen, and Marko Pettissalo. On the coverage of lpwans: range evaluation and channel attenuation model for lora technology. In *2015 14th International Conference on ITS Telecommunications (ITST)*, pages 55–59. IEEE, 2015.

[75] Ferran Adelantado, Xavier Vilajosana, Pere Tuset-Peiro, Borja Martinez, Joan Melia-Segui, and Thomas Watteyne. Understanding the limits of lorawan. *IEEE Communications magazine*, 55(9):34–40, 2017.

[76] Aloÿs Augustin, Jiazi Yi, Thomas Clausen, and William Mark Townsley. A study of lora: Long range & low power networks for the internet of things. *Sensors*, 16 (9):1466, 2016.

[77] Keith E Nolan, Wael Guibene, and Mark Y Kelly. An evaluation of low power wide area network technologies for the internet of things. In *2016 international wireless communications and mobile computing conference (IWCMC)*, pages 439–444. IEEE, 2016.

[78] Juha Petäjäjärvi, Konstantin Mikhaylov, Marko Pettissalo, Janne Janhunen, and Jari Iinatti. Performance of a low-power wide-area network based on lora technology: Doppler robustness, scalability, and coverage. *International Journal of Distributed Sensor Networks*, 13(3):1550147717699412, 2017.

[79] Juha Petäjäjärvi, Konstantin Mikhaylov, Rumana Yasmin, Matti Hämäläinen, and Jari Iinatti. Evaluation of lora lpwan technology for indoor remote health and wellbeing monitoring. *International Journal of Wireless Information Networks*, 24 (2):153–165, 2017.

[80] Pierre Neumann, Julien Montavont, and Thomas Noël. Indoor deployment of low-power wide area networks (lpwan): A lorawan case study. In *2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 1–8. IEEE, 2016.

[81] Davide Sartori and Davide Brunelli. A smart sensor for precision agriculture powered by microbial fuel cells. In *2016 IEEE Sensors Applications Symposium (SAS)*, pages 1–6. IEEE, 2016.

[82] LoRa Alliance. Lorawan specification. *LoRa Alliance*, pages 1–82, 2015.

[83] LoRa Alliance. Lorawan certified products. *Availible: https://www. lora-alliance. org/Products/Certified-Products. Accessed in*, 5:19, 2017.

[84] Martin C Bor, Utz Roedig, Thiemo Voigt, and Juan M Alonso. Do lora low-power wide-area networks scale? In *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, pages 59–67, 2016.

[85] Dmitry Bankov, Evgeny Khorov, and Andrey Lyakhov. On the limits of lorawan channel access. In *2016 International Conference on Engineering and Telecommunication (EnT)*, pages 10–14. IEEE, 2016.

[86] Orestis Georgiou and Usman Raza. Low power wide area network analysis: Can lora scale? *IEEE Wireless Communications Letters*, 6(2):162–165, 2017.

[87] Marco Centenaro, Lorenzo Vangelista, Andrea Zanella, and Michele Zorzi. Long-range communications in unlicensed bands: The rising stars in the iot and smart city scenarios. *IEEE Wireless Communications*, 23(5):60–67, 2016.

[88] Christopher Yang, Ou Liang, Santiago Ontanon, Weimao Ke, Helen Loeb, and Charlie Klauer. Predictive modeling with vehicle sensor data and iot for injury prevention. In *2018 IEEE 4th International Conference on Collaboration and Internet Computing (CIC)*, pages 293–298. IEEE, 2018.

[89] Claire Goursaud and Jean-Marie Gorce. Dedicated networks for iot: Phy/mac state of the art and challenges. 2015.

[90] George Margelis, Robert Piechocki, Dritan Kaleshi, and Paul Thomas. Low throughput networks for the iot: Lessons learned from industrial implementations. In *2015 IEEE 2nd world forum on internet of things (WF-IoT)*, pages 181–186. IEEE, 2015.

[91] Maarten Weyn, Glenn Ergeerts, Luc Wante, Charles Vercauteren, and Peter Hellinckx. Survey of the dash7 alliance protocol for 433 mhz wireless sensor communication. *International Journal of Distributed Sensor Networks*, 9(12):870430, 2013.

[92] Lluís Casals, Bernat Mir, Rafael Vidal, and Carles Gomez. Modeling the energy performance of lorawan. *Sensors*, 17(10):2364, 2017.

[93] Norman Abramson. The aloha system: another alternative for computer communications. In *Proceedings of the November 17-19, 1970, fall joint computer conference*, pages 281–285, 1970.

[94] Floris Van den Abeele, Jetmir Haxhibeqiri, Ingrid Moerman, and Jeroen Hoebeke. Scalability analysis of large-scale lorawan networks in ns-3. *IEEE Internet of Things Journal*, 4(6):2186–2198, 2017.

[95] Dong Geun Jeong and Wha Sook Jeon. Performance of an exponential backoff scheme for slotted-aloha protocol in local wireless environment. *IEEE transactions on vehicular technology*, 44(3):470–479, 1995.

[96] Leonard Kleinrock and Fouad Tobagi. Packet switching in radio channels: Part i-carrier sense multiple-access modes and their throughput-delay characteristics. *IEEE transactions on Communications*, 23(12):1400–1416, 1975.

[97] Tommaso Polonelli, Davide Brunelli, and Luca Benini. Slotted aloha overlay on lorawan-a distributed synchronization approach. In *2018 IEEE 16th international conference on embedded and ubiquitous computing (EUC)*, pages 129–132. IEEE, 2018.

[98] Lora. [online]. available:www.lora-alliance.org/.

[99] 3GPP. [3gpp tr 45.820 v13.1.0. “cellular system support for ultra-low complexity and low throughput internet of things (ciot), November 2015.

[100] Pilar Andres-Maldonado, Pablo Ameigeiras, Jonathan Prados-Garzon, Jorge Navarro-Ortiz, and Juan M Lopez-Soler. Narrowband iot data transmission procedures for massive machine-type communications. *IEEE Network*, 31(6):8–15, 2017.

[101] Kais Mekki, Eddy Bajic, Frederic Chaxel, and Fernand Meyer. A comparative study of lpwan technologies for large-scale iot deployment. *ICT Express*, 2018.

[102] Rashmi Sharan Sinha, Yiqiao Wei, and Seung-Hoon Hwang. A survey on lpwa technology: Lora and nb-iot. *Ict Express*, 3(1):14–21, 2017.

[103] Godfrey Anuga Akpakwu, Bruno J Silva, Gerhard P Hancke, and Adnan M Abu-Mahfouz. A survey on 5g networks for the internet of things: Communication technologies and challenges. *IEEE Access*, 6:3619–3647, 2018.

[104] Kais Mekki, Eddy Bajic, Frederic Chaxel, and Fernand Meyer. Overview of cellular lpwan technologies for iot deployment: Sigfox, lorawan, and nb-iot. In *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pages 197–202. IEEE, 2018.

[105] Carles Gomez, Juan Carlos Veras, Rafael Vidal, Lluís Casals, and Josep Paradells. A sigfox energy consumption model. *Sensors*, 19(3):681, 2019.

[106] Tommaso Polonelli, Davide Brunelli, Achille Marzocchi, and Luca Benini. Slotted aloha on lorawan-design, analysis, and deployment. *Sensors*, 19(4):838, 2019.

[107] Y-P Eric Wang, Xingqin Lin, Ansuman Adhikary, Asbjorn Grovlen, Yutao Sui, Yufei Blankenship, Johan Bergman, and Hazhir S Razaghi. A primer on 3gpp narrowband internet of things. *IEEE Communications Magazine*, 55(3):117–123, 2017.

[108] Mads Lauridsen, Rasmus Krigslund, Marek Rohr, and Germán Madueno. An empirical nb-iot power consumption model for battery lifetime estimation. In *Ieee Vts... Vehicular Technology Conference*, 2018.

[109] Liang Zhou, Dan Wu, Jianxin Chen, and Zhenjiang Dong. Greening the smart cities: Energy-efficient massive content delivery via d2d communications. *IEEE Transactions on Industrial Informatics*, 14(4):1626–1634, 2018.

[110] Olakunle Elijah, Tharek Abdul Rahman, Igbafe Orikumhi, Chee Yen Leow, and MHD Nour Hindia. An overview of internet of things (iot) and data analytics in agriculture: Benefits and challenges. *IEEE Internet of Things Journal*, 5(5): 3758–3773, 2018.

[111] Chun Yeow Yeoh, Abdullah bin Man, Qazi Mamoon Ashraf, and Ahmad Kamsani Samingan. Experimental assessment of battery lifetime for commercial off-the-shelf nb-iot module. In *2018 20th International Conference on Advanced Communication Technology (ICACT)*, pages 223–228. IEEE, 2018.

[112] Vitaly Petrov, Andrey Samuylov, Vyacheslav Begishev, Dmitri Moltchanov, Sergey Andreev, Konstantin Samouylov, and Yevgeni Koucheryavy. Vehicle-based relay assistance for opportunistic crowdsensing over narrowband iot (nb-iot). *IEEE Internet of Things journal*, 5(5):3710–3723, 2018.

[113] Seongjung Ha, Hyoungwon Seo, Youngjunn Moon, Dongjun Lee, and Junho Jeong. A novel solution for nb-iot cell coverage expansion. In *2018 Global Internet of Things Summit (GIoTS)*, pages 1–5. IEEE, 2018.

[114] Ansuman Adhikary, Xingqin Lin, and Y-P Eric Wang. Performance evaluation of nb-iot coverage. In *Vehicular Technology Conference (VTC-Fall), 2016 IEEE 84th*, pages 1–5. IEEE, 2016.

[115] 3GPP. 3GPP ts 36.321 v14.2.0. evolved universal terrestrial radio access(E-UTRA); medium access control(MAC) protocol specification, March 2017.

[116] LoRa SX1276. 868/915mhz rf transceiver module, 2019.

[117] Tian Wang, Md Zakirul Alam Bhuiyan, Guojun Wang, Md Arafatur Rahman, Jie Wu, and Jiannong Cao. Big data reduction for a smart city's critical infrastructural health monitoring. *IEEE Communications Magazine*, 56(3):128–133, 2018.

[118] Lte e-utra physical layer procedures, ts 36.213 v13.8.0, 2018.

[119] Stephanie Baker, Wei Xiang, and Ian Atkinson. Internet of things for smart healthcare: Technologies, challenges, and opportunities. *IEEE Access*, 5, 2017.

[120] SM Riazul Islam, Daehan Kwak, MD Humaun Kabir, Mahmud Hossain, and Kyung-Sup Kwak. The internet of things for health care: a comprehensive survey. *IEEE Access*, 3:678–708, 2015.

[121] Michele Magno, Tommaso Polonelli, Filippo Casamassima, Andres Gomez, Elisabetta Farella, and Luca Benini. Energy-efficient context aware power management with asynchronous protocol for body sensor network. *Mobile Networks and Applications*, 22(5):814–824, 2017.

[122] D. Brunelli, E. Farella, L. Rocchi, M. Dozza, L. Chiari, and L. Benini. Bio-feedback system for rehabilitation based on a wireless body area network. In *Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06)*, pages 5 pp.–531. IEEE, 2006.

[123] Luis Filipe, Florentino Fdez-Riverola, Nuno Costa, and António Pereira. Wireless body area networks for healthcare applications: Protocol stack review. *International Journal of Distributed Sensor Networks*, 11(10):213705, 2015.

[124] Shih-Hao Chang, Rui-Dong Chiang, Shih-Jung Wu, and Wei-Ting Chang. A context-aware, interactive m-health system for diabetics. *IT Professional*, 18(3): 14–22, 2016.

[125] Texas Instruments. CC1352, 2019. URL `http://www.ti.com/product/CC1352R`.

[126] Texas Instruments . CC2652, 2019. URL `http://www.ti.com/product/CC2652R`.

[127] Roberto Meattini, Simone Benatti, Umberto Scarcia, Daniele De Gregorio, Luca Benini, and Claudio Melchiorri. An semg-based human–robot interface for robotic hands using machine learning and synergies. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 8(7):1149–1158, 2018.

[128] Texas Instruments. WL1831, 2019. URL `http://www.ti.com/product/WL1831MOD`.

[129] Pycom. lopy4. URL `http://pycom.io/product/lopy4/`.

[130] José Dias and António Grilo. Lorawan multi-hop uplink extension. *Procedia computer science*, 130:424–431, 2018.

[131] Cypress. WICED, 2019. URL `https://www.cypress.com/products/wiced-software`.

[132] ST Microelectronics. LoRaWAN software expansion for STM32Cube, 2019. URL `https://www.st.com/en/embedded-software/i-cube-lrwan.html`.

[133] Razali Ngah and Yasser Zahedi. Uwb communications: Present and future. In *2016 IEEE Asia-Pacific Conference on Applied Electromagnetics (APACE)*, pages 373–378. IEEE, 2016.

[134] Patricia Martigne. Uwb for low data rate applications: technology overview and regulatory aspects. In *2006 IEEE International Symposium on Circuits and Systems*, pages 4–pp. IEEE, 2006.

[135] Kent D Colling and Philip Ciorciari. Ultra wideband communications for sensor networks. In *MILCOM 2005-2005 IEEE Military Communications Conference*, pages 2384–2390. IEEE, 2005.

[136] Jinyun Zhang, Philip V Orlik, Zafer Sahinoglu, Andreas F Molisch, and Patrick Kinney. Uwb systems for wireless sensor networks. *Proceedings of the IEEE*, 97 (2):313–331, 2009.

[137] RS Thoma, Ole Hirsch, Jürgen Sachs, and Rudolf Zetik. Uwb sensor networks for position location and imaging of objects and environments. In *The Second European Conference on Antennas and Propagation, EuCAP 2007*, pages 1–9. IET, 2007.

[138] Yuheng Liu, Chao Li, Yang He, Jing Wu, and Zhang Xiong. A perimeter intrusion detection system using dual-mode wireless sensor networks. In *2007 Second International Conference on Communications and Networking in China*, pages 861–865. IEEE, 2007.

[139] Xiaojing Huang, Eryk Dutkiewicz, Ricardo Gandia, and Darryn Lowe. Ultrawideband technology for video surveillance sensor networks. In *2006 4th IEEE International Conference on Industrial Informatics*, pages 1012–1017. IEEE, 2006.

[140] Jia Li and Timothy Talty. Channel characterization for ultra-wideband intravehicle sensor networks. In *MILCOM 2006-2006 IEEE Military Communications conference*, pages 1–5. IEEE, 2006.

[141] Fabrizo Granelli, Honggang Zhang, Xiaofei Zhou, and Stefano Marano. Research advances in cognitive ultra wide band radio and their application to sensor networks. *Mobile Networks and Applications*, 11(4):487–499, 2006.

[142] Lucian Stoica, Alberto Rabbachin, Heikki Olavi Repo, Teemu Sakari Tiuraniemi, and Ian Oppermann. An ultrawideband system architecture for tag based wireless sensor networks. *IEEE transactions on vehicular technology*, 54(5):1632–1645, 2005.

[143] V Mehta and MEI Zarki. Ban ultra wide band (uwb) based sensor network for civil infrastructure health monitoring,[in proc. 1st eur. workshop wireless sensor netw.(ewsn). *Berlin, Germany, Jan*, pages 19–21, 2004.

[144] *ASP013: The implementation of two-way ranging with the DW1000.* Decawave Limited, 2015.

[145] IEEE Computer Society. Amendment to ieee std. 802.15.4, ieee std. 802.15.4a-2007. URL `http://book.itep.ru/depository/zigbee/802_15_4a-2007.pdf`.

[146] AF Molisch, F Chin, Y Kim, M Welborn, and J Lampe. Uwb phy proposal for ieee 802.15. 4a alt-phy project. *IEEE document 802.15-05-0172-02-004a*, 2005.

[147] Lav Gupta, Raj Jain, and Gabor Vaszkun. Survey of important issues in uav communication networks. *IEEE Communications Surveys & Tutorials*, 18(2):1123–1152, 2015.

[148] Janis Tiemann and Christian Wietfeld. Scalable and precise multi-uav indoor navigation using tdoa-based uwb localization. In *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–7. IEEE, 2017.

[149] Ashutosh Dhekne, Ayon Chakraborty, Karthikeyan Sundaresan, and Sampath Rangarajan. Trackio: tracking first responders inside-out. In *16th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 19)*, pages 751–764, 2019.

[150] Dan Popescu, Cristian Dragana, Florin Stoican, Loretta Ichim, and Grigore Stamatescu. A collaborative uav-wsn network for monitoring large areas. *Sensors*, 18 (12):4202, 2018.

[151] P. D. Mitcheson, D. Boyle, G. Kkelis, D. Yates, J. A. Saenz, S. Aldhaher, and E. Yeatman. Energy-autonomous sensing systems using drones. In *2017 IEEE SENSORS*, pages 1–3, Oct 2017. doi: 10.1109/ICSENS.2017.8234092.

[152] Yuan Qin, David Boyle, and Eric Yeatman. Efficient and reliable aerial communication with wireless sensors. *IEEE Internet of Things Journal*, 6(5):9000–9011, 2019.

[153] Philipp Mayer, Michele Magno, Christoph Schnetzler, and Luca Benini. Embeduwb: Low power embedded high-precision and low latency uwb localization. In *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*, pages 519–523. IEEE, 2019.

[154] Pablo Corbalán and Gian Pietro Picco. Concurrent ranging in ultra-wideband radios: Experimental evidence, challenges, and opportunities. In *EWSN*, pages 55–66, 2018.

[155] DW1000 IEEE802. 15.4-2011 uwb transceiver. *Datasheet, Decawave*, 2016.

[156] Cristina Budaciu, Nicolae Botezatu, Marius Kloetzer, and Adrian Burlacu. On the evaluation of the crazyflie modular quadcopter system. In *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1189–1195. IEEE, 2019.

[157] Karalikkadan Ashhar, Cheong Boon Soh, and Keng He Kong. Wireless ultrawideband sensor network for gait analysis in rehabilitation clinics. In *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 1524–1529. IEEE, 2018.

[158] Dora Spenza, Michele Magno, Stefano Basagni, Luca Benini, Mario Paoli, and Chiara Petrioli. Beyond duty cycling: Wake-up radio with selective awakenings for long-lived wireless sensing systems. In *2015 IEEE Conference on Computer Communications (INFOCOM)*, pages 522–530. IEEE, 2015.

[159] Ricardo C Carrano, Diego Passos, Luiz CS Magalhaes, and Celio VN Albuquerque. Survey and taxonomy of duty cycling mechanisms in wireless sensor networks. *IEEE Communications Surveys & Tutorials*, 16(1):181–194, 2013.

[160] Rajeev Piyare, Amy L Murphy, Csaba Kiraly, Pietro Tosato, and Davide Brunelli. Ultra low power wake-up radios: A hardware and networking survey. *IEEE Communications Surveys & Tutorials*, 19(4):2117–2157, 2017.

[161] Infineon. Rf and microwave power detection with schottky diodes, version x.y, 2018. URL `http://tiny.cc/o1thlz`.

[162] Version DW1000 User Manual. 2.11.(decawave, 2017), 2019.

[163] Marko Malajner, Peter Planinšič, and Dušan Gleich. Uwb ranging accuracy. In *2015 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pages 61–64. IEEE, 2015.

[164] Ruqiang Yan, Xuefeng Chen, and Subhas Chandra Mukhopadhyay. *Structural health monitoring*. Springer, 2017.

[165] Roberto Marani, Massimiliano Nitti, Ettore Stella, and Tiziana D'Orazio. Monitoring of indoor environments by change detection in point clouds. In *2016 IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems (EESMS)*, pages 1–6. IEEE, 2016.

[166] T Addabbo, A Fort, M Mugnaini, E Panzardi, A Pozzebon, M Tani, and V Vignoli. A low cost distributed measurement system based on hall effect sensors for structural crack monitoring in monumental architecture. *Measurement*, 116: 652–657, 2018.

[167] Alberto Girolami, Davide Brunelli, and Luca Benini. Low-cost and distributed health monitoring system for critical buildings. In *2017 IEEE Workshop on Environmental, Energy, and Structural Monitoring Systems (EESMS)*, pages 1–6. IEEE, 2017.

[168] Aman Kansal, Jason Hsu, Sadaf Zahedi, and Mani B Srivastava. Power management in energy harvesting sensor networks. *ACM Transactions on Embedded Computing Systems (TECS)*, 6(4):32–es, 2007.

[169] Satish Nagarajaiah and Kalil Erazo. Structural monitoring and identification of civil infrastructure in the united states. *Structural Monitoring and Maintenance*, 3(1):51, 2016.

[170] Siham G Farag. Application of smart structural system for smart sustainable cities. In *2019 4th MEC International Conference on Big Data and Smart City (ICBDSC)*, pages 1–5. IEEE, 2019.

[171] P. D. Mitcheson, D. Boyle, G. Kkelis, D. Yates, J. A. Saenz, S. Aldhaher, and E. Yeatman. Energy-autonomous sensing systems using drones. In *2017 IEEE SENSORS*, pages 1–3, 2017.

[172] Yuan Qin, David Boyle, and Eric Yeatman. Radio diversity for heterogeneous communication with wireless sensors. In *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*, pages 955–960. IEEE, 2019.

[173] Hetal P Mistry and Nital H Mistry. Rssi based localization scheme in wireless sensor networks: A survey. In *2015 Fifth International Conference on Advanced Computing & Communication Technologies*, pages 647–652. IEEE, 2015.

[174] Andrew Mittleider, Brent Griffin, and Carrick Detweiler. Experimental analysis of a uav-based wireless power transfer localization system. In *Experimental Robotics*, pages 357–371. Springer, 2016.

[175] Juan M Arteaga, Lingxin Lan, Samer Aldhaher, George Kkelis, David C Yates, and Paul D Mitcheson. A multi-mhz ipt-link developed for load characterisation at highly variable coupling factor. In *2018 IEEE Wireless Power Transfer Conference (WPTC)*, pages 1–4. IEEE, 2018.

[176] Samer Aldhaher, Paul D Mitcheson, Juan M Arteaga, George Kkelis, and David C Yates. Light-weight wireless power transfer for mid-air charging of drones. In *2017 11th European Conference on Antennas and Propagation (EUCAP)*, pages 336–340. IEEE, 2017.

[177] Shuo Liu, Jianhui Su, and Jidong Lai. Accurate expressions of mutual inductance and their calculation of archimedean spiral coils. *Energies*, 12(10):2017, 2019.

[178] Carrick Detweiler, Michael Eiskamp, Brent Griffin, Jennifer Johnson, Jinfu Leng, Andrew Mittleider, and Elizabeth Basha. Unmanned aerial vehicle-based wireless charging of sensor networks. In *Wireless Power Transfer Algorithms, Technologies and Applications in Ad Hoc Communication Networks*, pages 433–464. Springer, 2016.

[179] Michele Magno, Federico Tombari, Davide Brunelli, Luigi Di Stefano, and Luca Benini. Multimodal video analysis on self-powered resource-limited wireless smart camera. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 3(2):223–235, 2013.

[180] Michele Magno, Federico Tombari, Davide Brunelli, Luigi Di Stefano, and Luca Benini. Multimodal abandoned/removed object detection for low power video surveillance systems. In *2009 Sixth IEEE international conference on advanced video and signal based surveillance*, pages 188–193. IEEE, 2009.

[181] Luca Negri, Mariagiovanna Sami, David Macii, and Alessandra Terranegra. Fsm–based power modeling of wireless protocols: the case of bluetooth. In *Proceedings of the 2004 international symposium on Low power electronics and design*, pages 369–374, 2004.

[182] Leila Makkaoui, Vincent Lecuire, and Jean-Marie Moureaux. Fast zonal dct-based image compression for wireless camera sensor networks. In *2010 2nd International Conference on Image Processing Theory, Tools and Applications*, pages 126–129. IEEE, 2010.

[183] Wen-Hsiung Chen, CH Smith, and Sam Fralick. A fast computational algorithm for the discrete cosine transform. *IEEE Transactions on communications*, 25(9): 1004–1009, 1977.

[184] Tom Torfs, Tom Sterken, Steven Brebels, Juan Santana, Richard van den Hoven, Vincent Spiering, Nicolas Bertsch, Davide Trapani, and Daniele Zonta. Low power wireless sensor network for building monitoring. *IEEE Sensors Journal*, 13(3):909–915, 2012.

[185] Davide Rossi, Antonio Pullini, Michael Gautschi, Igor Loi, Frank Kagan Gurkaynak, Philippe Flatresse, and Luca Benini. A- 1.8 v to 0.9 v body bias, 60 gops/w 4-core cluster in low-power 28nm utbb fd-soi technology. In *2015 IEEE SOI-3D-Subthreshold Microelectronics Technology Unified Conference (S3S)*, pages 1–3. IEEE, 2015.

[186] Hossam Osman, Waseim Mahjoup, Azza Nabih, and Gamal M Aly. Jpeg encoder for low-cost fpgas. In *2007 International Conference on Computer Engineering & Systems*, pages 406–411. IEEE, 2007.

[187] Tadashi Sakamoto and Tomohiro Hase. Software jpeg for a 32-bit mcu with dual issue. *IEEE Transactions on Consumer Electronics*, 44(4):1334–1341, 1998.

[188] Hsieh Hou. A fast recursive algorithm for computing the discrete cosine transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35(10):1455–1461, 1987.

[189] Christoph Loeffler, Adriaan Ligtenberg, and George S Moschytz. Practical fast 1-d dct algorithms with 11 multiplications. In *International Conference on Acoustics, Speech, and Signal Processing,*, pages 988–991. IEEE, 1989.

[190] Yukihiro Arai, Takeshi Agui, and Masayuki Nakajima. A fast dct-sq scheme for images. *IEICE TRANSACTIONS (1976-1990)*, 71(11):1095–1097, 1988.

[191] Noritsuna. Jpeg encoder for cortex-m. URL `https://github.com/noritsuna/JPEGEncoder4Cortex-M/`.

[192] Moodstocks. Jpeg encoder for cortex-m. URL `https://github.com/Moodstocks/jpec/`.

[193] Alain Hore and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *2010 20th international conference on pattern recognition*, pages 2366–2369. IEEE, 2010.

[194] Nikesh Gondchawar, RS Kawitkar, et al. Iot based smart agriculture. *International Journal of advanced research in Computer and Communication Engineering*, 5(6): 838–842, 2016.

[195] Weiguang Ding and Graham Taylor. Automatic moth detection from trap images for pest management. *Comput. Electron. Agric.*, 123(C):17–28, April 2016. ISSN 0168-1699. doi: 10.1016/j.compag.2016.02.003.

[196] Adrian Kaehler and Gary Bradski. *Learning OpenCV 3: computer vision in C++ with the OpenCV library.* " O'Reilly Media, Inc.", 2016.

[197] Zoran Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 2, pages 28–31. IEEE, 2004.

[198] Alessio Burrello, Francesco Conti, Angelo Garofalo, Davide Rossi, and Luca Benini. Work-in-progress: Dory: lightweight memory hierarchy management for deep nn inference on iot endnodes. In *2019 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ ISSS)*, pages 1–2. IEEE, 2019.

[199] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The cifar-10 dataset. *online: http://www. cs. toronto. edu/kriz/cifar. html*, 55, 2014.

[200] Taoufik Bouguera, Jean-François Diouris, Jean-Jacques Chaillout, Randa Jaouadi, and Guillaume Andrieux. Energy consumption model for sensor nodes based on lora and lorawan. *Sensors*, 18(7):2104, 2018.